REMARKS

Claims 1-39 have been canceled; and claims 40-79 have been newly added. The amendments to the claims are solely for the purpose of clarifying the subject matter of Applicants' invention, and are not intended to narrow the scope of any claims or for any purpose related to patentability. Applicants submit that none of the amendments to the claims enter any new matter into the application.

A substitute specification and abstract have been provided. The specification and abstract have been amended to correct for typographical and/or grammatical errors found in the specification and abstract as originally filed, and to place the disclosure in better form for U.S. practice. No new matter has been added by these amendments. Applicants respectfully submit that the substitute specification is in compliance with 37 C.F.R. § 1.125(b).

Applicants have requested that the Examiner review and approve the enclosed proposed corrections to FIGS. 11A, 11B, 16, 17 and 44, marked in red on the attached copies of same. No new matter is added by the drawing changes. Copies of the proposed corrections have been sent to the Official Draftsman under a separate paper pursuant to MPEP § 608.02(r).

Attached hereto is a marked-up version of the changes made to the specification and claims by the current amendment. The attached page is captioned **"Version With Markings To Show Changes Made."**

In view of the above, it is respectfully requested that these amendments now be entered, and that prosecution on the merits of this application now be initiated. If, however, for any reason the Examiner does not believe that such action can be taken at this time, it is respectfully requested that the Examiner telephone Applicants' attorney at (908) 654-5000 in order to overcome any additional objections that the Examiner

might have.

If there are any additional charges in connection with this requested amendment, the Examiner is authorized to charge Deposit Account No. 12-1095.

Respectfully submitted,

Dated: January 16, 2003

By ______________________
Joseph J. Opalach
Registration No.: 36,229
LERNER, DAVID, LITTENBERG,
KRUMHOLZ & MENTLIK, LLP
600 South Avenue West
Westfield, New Jersey 07090
(908) 654-5000
Attorneys for Applicants

409684_1.DOC

~~DESCRIPTION~~

INFORMATION PROCESSING SYSTEM AND METHOD

BACKGROUND OF THE INVENTION

~~Technical Field~~

[0001] The present invention relates to an information processing system, an information processing method, an information recording medium, and a program distributing medium, and particularly, to a system and a method for distributing an encryption processing key in a system involving ~~an~~ encryption processing. Particularly, the invention relates to an information processing system, an information processing method, an information recording medium, and a program distributing medium, which uses a tree-structured hierarchical key distributing system, reconstructs a hierarchical key distributing tree according to a distributing device to reduce the amount of data ~~quantity~~ contained in a distributing key block to thereby reduce ~~a~~ distributing message ~~quantity~~ size, relieve~~s~~ loads of a content key distribution or data distribution when various keys are renewed, and ~~can hold safety of data~~provide data safely.

~~Background Art~~

[0002] Recently, various software date (which will be hereinafter Recently, various software data (which will be hereinafter called contents) such as game programs, voice data, image data, and so on have been actively circulated through a network such as an internet, or storage media capable of being circulated such as a DVD, CD, etc. These ~~circulation~~ contents are ~~reproduced~~ reproducible ~~by reception of data~~ by a PC (Personal Computer)~~-owned by a user or game apparatus~~, or by mounting a memory medium, or are stored in a recording device within a recording and reproducing apparatus attached to a PC and the like. ~~, for example, a memory card, a hard disk and the like,~~

~~the contents being utilized by new reproducing from the stored medium.~~

[0003] Information apparatuses such as a video game apparatus, PC Information apparatuses such as a video game apparatus, PC and the like, have an interface for receiving the ~~circulation~~ contents from a network or for getting access to a DVD, CD and the like, and further have control means necessary for reproducing the contents, ~~and~~ along with RAM, ROM and the like used as a memory region for programs and data.

[0004] A user can reproduce v~~V~~arious contents such as music data, image data, or programs ~~are called from a memory medium by user's instructions from the information apparatus such as a game apparatus, PC and the like used as a reproducing apparatus or user's instructions through input means connected, and are reproduced though~~ through the information apparatuses or a display, a speaker and the like connected thereto.

[0005] ~~Many software c~~Contents, such as game programs, music data, image data and the like, are generally held in their distribution rights by owners and sales agents. Accordingly, in distribution of these contents, there is a predetermined ~~using~~ use limitation, that is, the use of ~~software~~ contents is granted only to ~~to only~~ proper users so that reproduction without permission is not ~~made~~ allowed. ~~That is, generally, the constitution taking security into consideration is employed.~~

[0006] One procedure for ~~realizing the limit of~~ limiting use to authorized users is ~~an~~ through encryption processing ~~of distributed contents~~. ~~Namely, f~~For example, various contents such as voice data, image data, game programs and the like are encrypted ~~through an internet or the like are distributed~~prior to distribution, and means for decrypting the encrypted contents ~~distributed~~, that is~~,~~ a decryption key, is given only to ~~to only~~ persons confirmed to be a proper user.

[0007] ~~Encrypted data can be returned to decrypted data that can be used by decrypting processing in accordance with the predetermined procedure.~~ Data ~~encrypting using a decryption key for decrypting processing, and a decrypting method, using an encrypted key for encryption processing of information as described have been heretofore well known~~ encryption and decryption using keys is ~~weel~~well known.

[0008] There are a variety of ~~kinds of forms of~~ data encrypting and decrypting methods using an encryption key and a decryption key, but there is, as one example therefor, a system called a ~~so-called~~ "common key encryption system." In the common key encryption system, ~~with~~ an encryption key ~~used for encrypting processing for data~~ and a decryption key ~~used for decrypting data~~ are made to be common~~,~~. ~~a~~ The common key (content key) ~~used for these encrypting processing and decrypting~~ is given to a proper user so as to eliminate ~~the~~ data access by an invalid user. ~~As a typical system of the system as described, there~~An illustration of a common key system is DES (Data Encryption Standard).

[0009] The encryption key and the decryption key ~~used for the encrypting processing and decrypting~~ as described above can be obtained by applying a unidirectional function such as a hash function on the basis of a pass-word or the like, for example. ~~The~~ As used herein, a unidirectional function ~~herein termed~~ is a function from which it is very difficult to obtain an input conversely from an output. For example, ~~the unidirectional function is applied with~~ a pass-word (determined by a user) is used as an input to the unidirectional function, and the encryption key and the decryption key are produced on the basis of the output. It is ~~substantially~~ nearly impossible, from the encryption key and the decryption key thus obtained, to conversely obtain ~~a~~ the pass-word ~~which is an original datum thereof~~.

[0010] ~~A system making processing by an encryption key used for encryption and processing by a decryption key used for decrypting different algorithm is a system so-called a~~ Another type of system is the "public key encryption system." The public key encryption system user, ~~is a method using~~ a public key ~~that can be used by an unspecific user~~for encryption. ~~in which with respect to an encrypted document for a specific individual, encrypting processing is carried out using a public key issued by the specific individual.~~ The document encrypted by the public key can be subjected to decrypting ~~processing merely~~ by a private key corresponding to the public key. ~~used for the encrypting processing.~~ The private key is owned ~~merely~~ by the individual who issued the public key, and the document encrypted by the public key can be decrypted ~~merely~~ by the individual having the private key (content key). A typical public key encryption system is ~~a~~ RSA (Rivest-Shamir-Adleman) encryption. ~~By making use of such an encryption system, there can be provided~~As such, it is possible to provide a system for enabling ~~decrypting~~ decryption of encrypted contents ~~merely for~~only by a proper user.

[0011] In the content distributing systems as described above, ~~employs many constitutions in which~~ contents are encrypted ~~and stored in the recording media such as a network, or DVD, CD and the like~~ to provide them ~~for~~ to users, and ~~to provide~~ a content key is provided for decrypting the encrypted contents for ~~only~~ use by a proper user. There is proposed a ~~constitution~~ variation in which a content key for preventing invalid copies of the content key itself is encrypted ~~to provide~~before being provided ~~it~~ to ~~a~~ the proper user, and ~~an~~ the encrypted content key is decrypted using a decryption key owned ~~by only~~ only by the proper user. ~~to enable using the content key.~~

[0012] The judgment whether or not a user is proper is generally carried out by executing authenticating processing before distribution of contents or content keys, for example, between a

content provider who is a transmitter of contents and a user's device. In general authenticating processing, confirmation is made of a mating party, and a session key effective only for communication is produced. When authentication is established, data, for example, contents or a content key, is encrypted using the produced session key for communication. The authenticating system includes mutual authentication using a common key encryption system, and an authentication system using a public key system. In the authentication using a common key, ~~a~~ the common key ~~in the~~ must be available system wide ~~is necessary,~~ which is inconvenient at the time of renewal processing. Further, in the public key system, the computation load is large ~~and necessary memory quantity increases, and t~~along with requiring larger amounts of memory. The provisioning of such a processing means on each device is not ~~a~~ desirable ~~constitution~~.

~~Disclosure~~ Summary of the Invention

[0013] It is an object of the present invention to provide an information processing system, an information processing method, an information recording medium, and a program distributing medium, which enables the safe transmission of data ~~safely~~ to a proper user without relying on the mutual authentication processing between a transmitter and a receiver of data as described above, and reconstructs a hierarchical key distribution tree according to a distribution tree in order to reduce the amount of data ~~quantity~~ contained in a distribution key block to thereby reduce ~~data quantity~~ the size of an encryption key, reduce~~s~~ the load of data transmission, and ~~enables reduction of~~reduce the processing requirements for obtaining an encryption key in each device.

[0014] An information processing system according to the present invention is one for distributing encrypted message data that is capable of only being used ~~only~~ in not less than one selected device. ~~selected, t~~The ~~individual~~ device comprising: encryption

processing means for holding a different key set of a node key, which is peculiar to each node in a hierarchical tree structure having a plurality of different devices as leaves, and a leaf key, which is peculiar to each device, and executing a decrypting process of ~~the~~ encrypted message data distributed to ~~a~~ the device using the key set; wherein the encrypted message data distributed to the device has data ~~constitution to be~~ encrypted with a renewal node key, which is obtained ~~in a~~ by decrypting ~~process of~~ an enabling key block (EKB). The enabling key block (EKB) includes a data part comprising encrypted key data, and a tag part, which is position discrimination data of the encrypted key data in the hierarchical tree structure. ~~Including~~ The EKB includes encrypted key data into which the renewal node key ~~into which~~ of at least one of the node keys in a group ~~constituted by~~ comprising nodes and leaves ~~connected at subordinate of a top node which is one node~~ of the hierarchical tree structure is encrypted by ~~the~~ a node key or ~~the~~ a leaf key in the group. ~~, and the enabling key block (EKB) includes a data part constituted by the encrypted key data and a tag part as position discrimination data of the encrypted key data in the hierarchical tree structure.~~

[0015] Further, in one embodiment of the information processing system according to the present invention, the encrypted key data ~~included in the enabling key block (EKB)~~ is data into which a node key ~~constituting~~ of the hierarchical tree structure is encrypted using a subordinate node key or a subordinate leaf key, and position discrimination data stored in the tag part ~~is constituted as~~ comprises a tag indicating whether there is ~~the~~ encrypted key data at a subordinate left and right node, or leaf position of a node. ~~position of each of not less than one encrypted key data stored in the enabling key block (EKB) or not.~~

[0016] Further, in one embodiment of the information processing system according to the present invention, the encrypted key data

~~included in the enabling key block (EKB) is constituted on the basis of~~ comprises only keys corresponding to a node or a leaf of a reconstructed hierarchical tree that is reconstructed by selecting paths constituting a simplified 2-branched type tree with terminal nodes or leaves with which the enabling key block (EKB) can be decrypted at the lowest stage to omit unnecessary nodes, and position discrimination data stored in the tag part includes data indicating whether the encrypted key corresponding to the tag of the enabling key block (EKB) is stored or not.

[0017] Further, in one embodiment of the information processing system according to the present invention, the encrypted key data comprises ~~included in the enabling key block (EKB) is constituted on the basis of only~~ a key corresponding to a node or a leaf of a reconstructed hierarchical tree that is reconstructed by selecting paths constituting a simplified 2-branched type tree with terminal nodes or leaves with which the enabling key block (EKB) can be decrypted at the lowest stage to omit unnecessary nodes, and position discrimination data stored in the tag part includes tags for indicating whether there is encrypted key data at a left and a right node or a leaf position at a subordinate ~~of a~~ node ~~position of each of not less than one encrypted key data stored in the enabling key block (EKB)~~, and data for indicating whether the encrypted key corresponding to the tag is stored or not.

[0018] Further, in one embodiment of the information processing system according to the present invention, the reconstructed hierarchical tree is a tree constituted by selecting a sub-root, which is a top node of an entity defined as a subset tree of devices having a common element.

[0019] Further, in one embodiment of the information processing system according to the present invention, the encrypted key data comprises, ~~included in the enabling key block (EKB) is constituted,~~ (in a simplified multi-branched type tree having a

terminal node or a leaf with which the enabling key block (EKB) can be decrypted at the lowermost stage~~,~~) ~~on the basis of only~~ keys corresponding to a top node and terminal nodes or leaves, of a reconstructed hierarchical tree that is reconstructed by selecting paths directly connecting the terminal nodes or leaves and a top of the multi-branched type tree to omit an unnecessary node, and position discrimination data stored in the tag part that includes data indicating whether an encrypted key corresponding to the tag of the enabling key block (EKB) is stored or not.

[0020] Further, in one embodiment of the information processing system according to the present invention, the reconstructed hierarchical tree is a tree having not less than three branches connecting the top node ~~constituting~~ (of ~~the~~ a simplified multi-branched type tree) with terminal nodes or leaves ~~constituting the simplified tree directly~~.

[0021] Further, in one embodiment of the information processing system according to the present invention, the encryption processing means in the device ~~has a constitution for~~ sequentially ~~extracting~~ extracts the encrypted key data with data of the tag part in the enabling key block (EKB), ~~executing~~ executes a decrypting process to obtain the renewal node key, and ~~executing decryption~~ decrypts ~~of~~ the encrypted message data with the obtained renewal node key ~~obtained~~.

[0022] Further, in one embodiment of the information processing system according to the present invention, the message data is a content key that can be used as a decryption key for decrypting content ~~data~~.

[0023] Further, in one embodiment of the information processing system according to the present invention, the message data is an authentication key used in the authentication process.

[0024] Further, in one embodiment of the information processing system according to the present invention, the message data is a

key for generating an integrity check value (ICV) of the content.

[0025] Further, in one embodiment of the information processing system according to the present invention, the message data is ~~a~~ program code.

[0026] Further, an information processing method according to Further, an information processing method according to the present invention is one for distributing encrypted message data capable of, only being used ~~only~~ in not less than one selected device~~s,~~. The method comprising: an enabling key block (EKB) generating step ~~of~~ for generating an enabling key block (EKB) ~~including~~ comprising a data part including encrypted key data into which the renewal node key ~~into which~~ of at least one of the node keys in a group ~~constituted by~~ comprising, nodes and leaves ~~connected at subordinate of a top node which is one node~~ of the hierarchical tree structure is renewed is encrypted with a node key or a leaf key in the group, and a tag part, which is position discrimination data in the hierarchical tree structure of encrypted key data stored in the data part; and a message data distribution step for generating message data encrypted with the renewal node key to distribute it to a device.

[0027] Further, one embodiment of the information processing method according to the present invention comprises a decrypting processing step of executing a decrypting process ~~to~~ on the encrypted message data using the key set in a device holding a different key set of a node key, which is peculiar to each node in the hierarchical structure, and a leaf key peculiar to each device.

[0028] Further, in one embodiment of the information processing method according to the present invention, the enabling key block (EKB) generating step includes a step of encrypting a node key ~~constituting~~ of the hierarchical tree structure using a subordinate node key, or a subordinate leaf key, to generate the encrypted key data, and a step of generating a tag indicating

whether there is encrypted key data at a node, or leaf position, at subordinate left and right positions of a node position. ~~of each of not less than one encrypted key data stored in the enabling key block (EKB) or not to store it in the tag part.~~

[0029] Further, in one embodiment of the information processing method according to the present invention, the enabling key block (EKB) generating step includes a step of generating a reconstructed hierarchical tree by selecting paths ~~constituting~~ of a simplified 2-branched type tree with a terminal node or leaf capable of decrypting the enabling key block (EKB) at the lowest stage to omit unnecessary nodes; a step of generating an enabling key bock (EKB) ~~on the basis of~~ using only a key corresponding to a ~~constitution~~ node or leaf of the reconstructed hierarchical tree; and a step of storing data indicating whether an encrypted key corresponding to a tag of the enabling key block (EKB) is stored in the tag part or not.

[0030] Further, in one embodiment of the information processing method according to the present invention, the step of generating the reconstructed hierarchical tree ~~is~~ includes a tree generating processing executed by selecting a sub-root, which is a top node of an entity defined as a subset tree of devices having a common element.

[0031] Further, in one embodiment of the information processing method according to the present invention, the enabling key block (EKB) generating step includes a step of generating~~,~~ (in the simplified branched type tree with a terminal node, or leaf, capable of decrypting the enabling key bock (EKB) at the lowest stage)~~,~~ the reconstructed hierarchical tree ~~reconstructed~~ by selecting a path for directly connecting the terminal node, or leaf, with the top of the multi-branched type tree; and a step of storing data indicating whether an encrypted key (corresponding to a tag of the enabling key bock (EKB)) is stored in the tag part or not.

[0032] Further, in one embodiment of the information processing method according to the present invention, the reconstructed hierarchical tree generated ~~in the step of generating the reconstructed hierarchical tree~~ is generated as a tree having not less than three branches ~~having~~ connecting a top node ~~constituting~~ (of a simplified multi-branched type tree) and a terminal node, or leaf. ~~constituting a simplified tree connected directly.~~

[0033] Further, in one embodiment of the information processing method according to the present invention, the decrypting processing step includes a renewal node key obtaining step ~~of~~ for obtaining the renewal node key by sequentially extracting encrypted key data stored in the data part on the basis of position discrimination data stored in the tag part of the enabling key block (EKB) ~~to sequentially execute decrypting process~~; and a message data decrypting step for executing decryption of the encrypted message data with the renewal node key.

[0034] Further, in one embodiment of the information processing method according to the present invention, the message data is a content key capable of being used as a decryption key for decrypting the content data.

[0035] Further, in one embodiment of the information processing method according to the present invention, the message data is an authentication key used in the authentication process.

[0036] Further, in one embodiment of the information processing method according to the present invention, the message data is a key ~~of~~ for generating an integrity check value (ICV) of contents.

[0037] Further, in one embodiment of the information processing method according to the present invention, the message data is ~~a~~ program code.

[0038] Further, an information recording medium according to the present invention ~~is one having data stored. The recording medium~~ stores an enabling key block (EKB). ~~Including~~The EKB comprises a data part, including encrypted key data into which the renewal node key ~~into which~~ of at least one of the node keys in a group ~~constituted by~~ comprising nodes and leaves ~~connected under a top node which is one node~~ of the hierarchical tree structure ~~is renewed~~ is encrypted with a node key or a leaf key in the group, and a tag part, which is position discrimination data in the hierarchical tree structure of encrypted key data stored in the data part, and message data encrypted by the renewal node key.

[0039] Further, in one embodiment of the information recording medium according to the present invention, the encrypted key data included in the enabling key block (EKB) is data into which the node key ~~constituting~~ of the hierarchical tree structure is encrypted using a subordinate node key or a subordinate leaf key; and the position discrimination data stored in the tag part is ~~constituted as~~ a tag indicating whether there is key data at the node, or of leaf, position at the subordinate left and right positions of the node position. ~~of each of not less one encrypted key data stored in the enabling key block (EKB).~~

[0040] Further, in one embodiment of the information recording medium according to the present invention, the encrypted key data ~~included in the enabling key block (EKB) is constituted on the basis of only~~ comprises a key corresponding to a node, or a leaf, of a reconstructed hierarchical tree that is reconstructed by selecting paths ~~constituting~~ of a simplified 2-branched type tree with a terminal node, or leaf, capable of decrypting the enabling key block (EKB) at the lowest stage to omit unnecessary nodes; and the position discrimination data stored in the tag part includes data indicating whether an encrypted key corresponding to the tag of the enabling key block (EKB) is stored or not.

[0041] A program distributing medium according to the present invention is one for distributing a computer program to execute on a computer system a process of generating an enabling key block (EKB) into which a renewal node key ~~into which~~ of at least one of the node keys in a group ~~constituted by~~ comprising nodes and ~~a~~ leaves ~~connected under the top node which is one node~~ of the hierarchical tree structure ~~is renewed~~ is encrypted with a node key or a leaf key in the group. The computer program includes a step of generating a reconstructed hierarchical tree by selecting a path ~~constituting~~ of a simplified 2-branched type tree with a terminal node, or a leaf, capable of decrypting the enabling key block (EKB) at the lowest stage to omit an unnecessary node; a step of generating the enabling key block (EKB) on the basis of only a key corresponding to a ~~constitution~~ node or leaf of the reconstructed hierarchical tree; and a step of storing data indicating whether an encrypted key corresponding to a tag of the enabling key block (EKB) is stored or not.

[0042] In ~~the constitution of~~ one aspect of the present invention, ~~the~~ distribution of an encryption key ~~distributing constitution of the~~ in accordance with a hierarchical tree ~~structure of the tree structure~~ is used to suppress the distributing message quantity necessary for key renewal as small as possible. That is, the key distribution method in which each apparatus~~es~~ is arranged in each leaf by n-division is used ~~whereby~~ to distribute, for example, a content key, which is an encryption key of content data, or an authentication key used in authentication processing or a program code ~~are distributed~~ along with an enabling key block through recording medium or a communication circuit.

[0043] Further, the enabling key block ~~is constituted by~~ comprises an encrypted key data part and a tag part, ~~showing~~ which shows a position of the encrypted key, whereby the amount of data ~~quantity~~ is reduced to enable rapid execution of a

decrypting process~~ing~~ in a device. ~~According to the present constitution,~~In accordance with an aspect of the invention, only the proper device is able to distribute decodable data safely.

[0044] It is noted It is noted that the program distributing medium according to the present invention is a medium for distributing a computer program in the form that can be read by a computer to a general computer system capable of executing, for example, various program codes. The medium includes recording media such as CD, FD, MO, etc., or a transfer medium such as a network, whose form is not particularly limited.

[0045] Such a program distributing medium defines a cooperative relationship ~~in terms of constitution or function~~ between a computer program and a distributing medium. ~~in order to realize a function of a predetermined computer program in a computer system.~~ In other words, a computer program is installed in a computer system through the distributing medium to exhibit the cooperative operation in the computer system to obtain the operation and effects described herein. ~~similar to another aspects.~~

[0046] The other objects, features and advantages of the present invention will be apparent from the detailed description with reference to the embodiments and the accompanying drawings of the present invention.

Brief Description of the Drawings

[0047] FIG. 1 is a view ~~for explaining~~ of an example ~~of constitution~~ of an information processing system according to the present invention.

[0048] FIG. 2 is a block diagram showing an example of ~~constitution of~~ a recording and reproducing apparatus that can be applied in the information processing system according to the present invention.

[0049] FIG. 3 ~~is a~~ shows an illustrative tree ~~constitution view~~ for use in explaining the encryption processing of various keys

and data in the information processing system according to the present invention.

[0050] FIGS. 4A and 4B are views each showing an example of an enabling key block (EKB) used in the distribution of various keys and data in the information processing system according to the present invention.

[0051] FIG. 5 is a view showing an example of distribution and ~~an example of~~ decrypting ~~processing~~ using an enabling key block ~~(EKB) of content keys~~ in the information processing system according to the present invention.

[0052] FIG. 6 is a view showing an ~~example of a~~ illustrative format of an enabling key block (EKB) in the information processing system according to the present invention.

[0053] FIGS. 7A to 7C are views ~~each for explaining a constitution of~~ illustrating a tag of an enabling key block (EKB) in the information processing system according to the present invention.

[0054] FIGS. 8A and 8B are views ~~each showing~~ illustrating an enabling key block (EKB) ~~and an example of data constitution for distributing~~ and the distribution of content keys and contents in the information processing system according to the present invention.

[0055] FIG. 9 is a view showing an example of processing in a device ~~in case of distributing~~ with respect to an enabling key block (EKB), content keys, and contents in the information processing system according to the present invention.

[0056] FIG. 10 is a view ~~for explaining the situation how to cope with~~ illustrating the case where an enabling key block (EKB) and contents are stored in the information processing system according to the present invention.

[0057] FIGS. 11A and 11B ~~are views each showing~~ illustrate a comparison between processing ~~for sending an enabling key block (EKB) and contents~~ in the information processing system according

to the present invention and ~~a~~ conventional ~~sending~~ processing.

[0058] FIG. 12 is a view showing an authentication processing sequence according to an applicable common key encryption system in the information processing system according to the present invention.

[0059] FIG. 13 is a view ~~(1)~~ showing an enabling key block (EKB), ~~a~~ data ~~constitution for distributing~~ distribution with an authentication key, and ~~a~~ processing ~~example~~ by a device in the information processing system according to the present invention.

[0060] FIG. 14 is ~~a~~ another view ~~(2)~~ showing an enabling key block (EKB), ~~a~~ data ~~constitution for distributing~~ distribution with an authentication key, and ~~a~~ processing ~~example~~ by a device in the information processing system according to the present invention.

[0061] FIG. 15 is a view showing an authentication processing sequence by a public key encryption system applicable in the information processing system according to the present invention.

[0062] FIG. 16 is a view showing ~~a~~ processing for distributing an enabling key block (EKB) and content keys using the authentication principle by a public key encryption system in the present invention.

[0063] FIG. 17 is a view showing ~~a~~ processing for distributing an enabling key block (EKB) and encrypted program data in the information processing system according to the present invention.

[0064] FIG. 18 is a view showing an example of MAC value production used in production of a content integrity check value (ICV) applicable in the present invention.

[0065] FIG. 19 is a view ~~(1)~~ showing ~~a data constitution for distributing~~ distribution of an enabling key block (EKB) and an ICV producing key, and ~~an example of a~~ illustrating processing in a device in the information processing system according to the present invention.

[0066] FIG. 20 is ~~a~~ another view ~~(2)~~ showing distribution of ~~a data constitution for distributing~~ an enabling key block (EKB) and an ICV producing key, and ~~an example of a~~ ~~illuatrative~~ illustrative processing in a device in the information processing system according to the present invention.

[0067] FIGS. 21A and 21B are views ~~each for~~for use in explaining a copy preventive function where an applicable content integrity check value (ICV) is stored in a medium in the present invention.

[0068] FIG. 22 is a view for ~~explaining a constitution for controlling~~ illustrating the control of an applicable content integrity check value (ICV) separately from a content storage medium in the present invention.

[0069] FIG. 23 is a view ~~for explaining an example of category classification of~~ illustrating a hierarchical tree structure in the information processing system of the present invention.

[0070] FIGS. 24A and 24B are views ~~each~~ for use in explaining ~~a~~ the production ~~producing process~~ of a simplified enabling key block (EKB) in the information processing system of the present invention.

[0071] FIGS. 25A and 25B are views ~~each~~ for use in explaining ~~a~~ the production ~~producing process~~ of an enabling key block (EKB) in the information processing system of the present invention.

[0072] FIGS. 26A and 26B are views ~~each~~ for use in explaining a simplified enabling key block (EKB) ~~(Example 1)~~ in the information processing system of the present invention.

[0073] FIGS. 27A and 27B are additional views ~~each~~ for use in explaining a simplified enabling key block (EKB) ~~(Example 2)~~ in the information processing system of the present invention.

[0074] FIGS. 28A to 28C are views ~~each~~ for use in explaining ~~an~~ entity control ~~constitution~~ of a hierarchical tree structure in the information processing system of the present invention.

[0075] FIGS. 29A to 29C are views ~~each~~ for use in explaining, in detail, ~~an~~ entity control ~~constitution of a hierarchical tree~~

~~structure~~ in the information processing system of the present invention.

[0076] FIGS. 30A and 30B are additional views ~~each~~ for use in explaining ~~an~~ entity control ~~constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0077] FIG. 31 is a view for use in explaining a reserve node ~~in an entity control constitution~~ of a hierarchical tree structure in the information processing system of the present invention.

[0078] FIG. 32 is a view for use in explaining a new entity registration sequence ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0079] FIG. 33 is a view for use in explaining a relationship between a new entity and a host entity ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0080] FIGS. 34A and 34B are views ~~each~~ for use in explaining a sub-EKB ~~used in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0081] FIGS. 35A to 35D are views ~~each~~ for use in explaining ~~a~~ device revoke processing ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0082] FIG. 36 is ~~a~~ another view for use in explaining ~~a~~ device revoke processing ~~sequence in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0083] FIGS. 37A and 37B are views ~~each~~ for use in explaining a renewal sub-EKB at the time of a device ~~revoke~~ revocation ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0084] FIGS. 38A to 38D are views ~~each~~ for use in explaining ~~an~~ entity revoke processing ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0085] FIG. 39 is ~~a~~ another view for use in explaining ~~an~~ entity revoke processing ~~sequence in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0086] FIG. 40 is a view ~~for explaining~~ illustrating a relationship between a revoke entity and a host entity ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0087] FIG. 41 is a view for use in explaining ~~a~~ capability setting ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0088] FIG. 42 is ~~a~~ another view for use in explaining ~~a~~ capability setting ~~in an entity control constitution of a hierarchical tree structure~~ in the information processing system of the present invention.

[0089] FIGS. 43A and 43B are views ~~each for explaining~~illustrating a capability control table for controlling a key issuing center (KDC) in the information processing system of the present invention.

[0090] FIG. 44 ~~is an~~shown an illustrative EKB producing processing flowchart ~~on the basis of a capability control table for controlling a key issuing center (KDC)~~ in the information processing system of the present invention.

[0091] FIG. 45 is a view ~~for explaining a~~ illustrating capability notice processing ~~at the time of new entity registration~~ in the information processing system of the present invention.

~~Best mode for Carrying out the Invention [Outline of~~

~~System]~~Detailed Description

[0092] FIG. 1 shows an example of a content distributing system to which the data processing system of the present invention can be applied. The content distributing side 10 transmits ~~a~~ encrypted content, or ~~a~~ an encrypted content key, ~~encrypted~~ to various content reproducible apparatuses on the content receiving side 20. The apparatus on the content receiving side 20 decrypts ~~an~~ the received encrypted content or ~~a~~ the received encrypted content key, ~~received~~ to obtain ~~a~~ the content, or ~~a~~ the content key, and carries out reproduction of image data and voice data or execution of various programs. The exchange of data between the content distributing side 10 and the content receiving side 20 is executed through a network such as an internet or through a circulatable recording medium such as DVD, CD.

[0093] The data distributing means on the content distributing side 10 includes an internet 11, a broadcasting satellite ~~broadcasting~~ 12, a telephone circuit 13, media 14 such as DVD, CD, etc., and on the other hand, the devices on the content receiving side 20 include a personal computer (PC) (21 or 22) portable apparatuses 23 such as a portable device (PD), a portable telephone, PDA (Personal Digital Assistants), etc., a recording and reproducing unit 24 such as DVD, CD players, and a reproduction exclusive-use unit 25 such as a game terminal. In these devices on the content receiving side 20, contents distributed from the content distributing side 10 are obtained from communication means such as a network, or from a media 30.

~~[Constitution of Device]~~

[0094] FIG. 2 shows a block diagram of a recording and reproducing device 100 as one example of devices on the content receiving side 20 shown in FIG. 1. The recording and reproducing device 100 has an input/output I/F (Interface) 120, a MPEG (Moving Picture Experts Group) codec 130, an I/F (Interface) 140 provided with A/D, D/A converter 141, an encryption processing

means 150, ROM (Read Only Memory) 160, CPU (Central Processing Unit) 170, a memory 180, and a drive 190 for a recording medium 195, which are connected to each other by a bus 110.

[0095] The input/output I/F 120 receives a digital signal ~~constituting~~ comprising various contents such as an image, voice, a program, etc., ~~supplied from the outside to output it~~ and provide the content to the bus 110, and, conversely, receives a digital signal ~~of~~ from the bus 110 ~~to output~~ and provides it to the outside. The MPEG codec 130 decrypts MPEG coded data supplied through the bus 110 to output it to the input/output I/F 140, and MPEG-decrypts a digital signal supplied from the input/output I/F 140 to output it to the bus 110. The input/output I/F 140 contains an A/D, D/A converter 141 therein. The input/output I/F 140 receives an analog signal ~~as a~~ representing content supplied from the outside, which is subjected to A/D (Analog Digital) conversion by the A/D, D/A converter 141 whereby the signal is output as a digital signal to the MPEG codec 130. Conversely, ~~and~~ a digital signal from the MPEG codec 130 is subjected to D/A (Digital Analog) conversion by the A/D, D/A converter 141, which is output as an analog signal to the outside.

[0096] The encryption processing means 150 ~~is constituted~~ comprises ~~form~~, for example, ~~one~~ an LSI (Large Scale Integrated circuit) chip ~~LSI (Large Scale Integrated circuit)~~, ~~to execute~~ for performing encrypting, decrypting ~~processing~~ or authentication processing of a digital signal ~~as a content~~ supplied through the bus 110, and ~~output~~ for providing encrypted data and decrypted data to the bus 110. The encryption processing means 150 can be also realized by not only the one chip LSI but by a combination of various ~~soft wares or hard wares. The constitution of the processing means formed from the software configuration will be described later.~~ software and/or hardware.

[0097] ROM 160 stores program data processed by the recording and reproducing device. The CPU 170 executes programs stored in

the ROM 160 and the memory 180 to thereby control the MPEG codec 130 and the encryption processing means 150. The memory 180 is for example, a non-volatile memory, which stores a program that is executed by the CPU 170, data necessary for operation of CPU 170, and a key set used in the encryption processing executed by the device. The key set will be explained later. The drive 190 drives the recoding medium 195 capable of recording and reproducing digital data to thereby read (reproduce) digital data from the recording medium 195 to output it to the bus 110, and supplies digital data supplied through the bus 110 to the recording medium 195 for recording.

[0098] The recording medium 195 is a medium capable of storing digital data, for example, an optical disk such as DVD, CD, an optical magnetic disk, a magnetic disk, a magnetic tape, or a semiconductor memory such as RAM, and in the present embodiment, the medium can be detachably mounted on the drive 190. However, the recording medium 195 may be housed in the recording and reproducing device 100.

[0099] The encryption processing means 150 shown in FIG. 2 may ~~be constituted as~~ comprise a single one-chip LSI, and may ~~employ a constitution that is~~ also be realized by a combination of ~~a~~ software and a hardware.

~~[Tree structure as a key distributing constitution]~~

[0100] Next, ~~the constitution~~ an arrangement for holding an encryption processing key in each device and a data distributing ~~constitution~~ arrangement where encrypted data are distributed from the content distributing side 10 shown in FIG. 1 to each device on the content receiving side 20 will be described using FIG. 3.

[0101] Numbers 0 to 15 shown in the lowest stage in FIG. 3 are individual devices on the content receiving side 20. That is, each leaf of the hierarchical tree structure shown in FIG. 3 corresponds to a device.

[0102] Each of devices 0 to 15 stores a key set comprising ~~a~~ the keys assigned to ~~a~~ each node from its own leaf to a root (~~a~~ node keys) and ~~a~~ its leaf key ~~of each leaf~~, in the hierarchical tree shown in FIG. 3~~,~~. This key set is determined at the time of manufacture or at the time of shipment, or afterwards. K0000 to K1111 shown in the lowest stage of FIG. 3 are respectively leaf keys assigned to devices 0 to 15, and keys from KR to K111 described in the second node from the lowest stage are node keys.

[0103] ~~In the constitution shown in FIG. 3, f~~For example, a device 0 has a key set comprising a leaf key K0000 and node keys K000, K00, K0, KR. A device 5 has a key set comprising K0101, K010, K01, K0, KR. A device 15 has a key set comprising K1111, K111, K11, K1, KR. In the tree of FIG. 3, only 16 devices (0 to 15) are described, and the tree structure ~~is shown as a systematic constitution to left and right~~ illustrates a well balanced ~~of~~ a 4-stage ~~constitution~~ tree. However, ~~much~~ many more devices may be ~~constituted~~ accommodated in ~~the~~ a tree, and the parts of ~~the~~ a tree may have ~~the~~ different numbers of stages.

[0104] Further, each device included in the tree structure shown in FIG. 3 includes various recording media, for example, DVD, CD, MD of the embedded type or the type detachably mounted on the device, or devices of various types using a flash memory or the like. Further, various application services may coexist. In ~~addition to the coexisting constitution of various devices and various application~~ this context, the hierarchical tree structure ~~which is a content or a key distributing constitution~~ shown in FIG. 3 is applied.

[0105] In the system in which various devices and applications coexist, for example, a portion surrounded by the dotted line in FIG. 3, that is, the devices 0, 1, 2 and 3 are illustratively set as a single group using the same recording medium. For example, with respect to the device included in the group surrounded by the dotted line, processing is executed such that ~~a~~ common

content is encrypted and sent from a provider, a content key used in common to devices is sent, or payment data for content charges is also encrypted and output from each device to a provider or a settlement organization. ~~The~~ Similarly, an organization (such as a content provider or a settlement organization) for carrying out data ~~transmit-receiving~~transmission to and from the devices ~~such as a content provider or a settlement organization~~ executes processing ~~for sending the portion surrounded by the dotted line of FIG. 3, that is, data collectively with~~treating the devices 0, 1, 2, 3 as one group. A plurality of such groups are present in the tree of FIG. 3. The organization ~~for carrying out data transmit-receiving to and from devices such as a content provider or a settlement organization~~ functions as a message data distributing means.

[0106] Node keys and leaf keys may be collectively controlled ~~collectively~~ by a single key control center, or may be controlled ~~every~~ on a group basis by the message data distributing means ~~such as a provider, or a settlement organization for carrying out transmit-receiving of various data with respect to groups~~. These node keys and leaf keys are subjected to renewal processing when a key is leaked. This renewal processing is executed by a key control center, a provider or a settlement organization.

[0107] In this tree structure, as will be apparent from FIG. 3, three devices 0, 1, 2~~,~~ and 3 included in one group hold common node keys K00, K0, KR. ~~as a node key.~~ By utilizing ~~this~~ these common node keys ~~common constitution~~, for example, a common content key can be distributed to only devices 0, 1, 2, 3. For example, if the node key K00 ~~itself held in common~~ is set as a content key, only the devices 0, 1, 2, 3 can ~~be set as~~ utilize key K00 as a common content key. ~~without executing new sending of key.~~ Further, if encrypted data ~~a value~~ Enc(K00, Kcon) ~~obtained by encrypting a new content key Kcon by a node key K00~~ is distributed to the devices 0, 1, 2, 3 through a network or by

being stored in the recording medium, only the devices 0, 1, 2, 3 can decrypt~~ion~~ the encrypted data Enc(K00, Kcon) using ~~a~~ the common node key K00 ~~held in the respective devices~~ to obtain ~~a~~ the content key: Kcon. (As used herein, t~~T~~he notation Enc(Ka, Kb) indicates data into which Kb is encrypted by Ka.)

[0108] Further, where at the time t, keys : K0011, K001, K00, K0~~,~~ and KR owned by the device 3 are analyzed by a hacker and then exposed, it is necessary for protecting subsequent data ~~transmit-received~~transmission to the group in ~~a system (a group of devices 0, 1, 2, 3)~~ to separate out the device 3 from the ~~system~~ group. To this end, node keys: K001, K00, K0, KR are respectively renewed to new keys K(t)001, K(t)00, K(t)0, K(t)R, which renewed keys ~~to be notified~~are sent to the devices 0, 1, 2. (As used ~~hierein~~herein, ~~Here,~~ K(t)aaa indicates a renewal of key ~~of~~ Kaaa ~~of~~ at time ~~generation :~~ t.)

[0109] The distributing ~~processing of~~of a renewal key will now be described. Renewal of a key is executed by storing a table ~~constituted~~ comprising a ~~by~~ block of data called ~~an~~ "enabling key block (EKB)"~~:~~ ~~Enabling Key Block) shown in FIG. 4A~~ in a network, for example, or in a recording medium ~~to~~ for supply ~~them~~ to the devices 0, 1 ~~,~~and 2. The enabling key block (EKB) ~~is constituted by~~ comprises a decryption key for distributing a newly renewed key ~~newly renewed~~ to a device corresponding to each leaf ~~constituting a~~ of the tree structure ~~as~~ shown in FIG. 3. The enabling key block (EKB) is sometimes called a key renewal block (KRB: Key Renewal Block).

[0110] In the enabling key block (EKB) shown in FIG. 4A, only ~~the device in which a node key need~~ those keys that need to be renewed ~~is constituted as block data having a data constitution that can be renewed. An example of FIGS. 4A and 4B shows, in the devices 0, 1 and 2 in the tree structure shown in FIG. 3, block data formed for the purpose of distributing a renewal node key of generation t.~~ comprise the EKB. As will be apparent from FIG. 3,

the device 0 and the device 1 require K(t)00, K(t)0, K(t)R as renewal node keys, and the device 2 requires K(t)001, K(t)00, K(t)0, K(t)R as renewal node keys.

[0111] As shown in ~~EKB of~~ FIG. 4A, a plurality of encrypted keys are included in the EKB. The encrypted key in the lowest stage is Enc(K0010, K(t)001). This is a renewal node key K(t)001 encrypted by a leaf key K0010 of the device 2, and the device 2 is able to decrypt this encrypted key by its leaf key to obtain K(t)001. By using K(t)001 obtained by decrypting, an encrypted key Enc(K(t)001, K(t)00) in the second stage from the bottom can be decrypted to obtain a renewal node key K(t)00. Sequentially, an encrypted key Enc(K(t)00, K(t)0) in the second stage from the top of the EKB of FIG. 4A is decrypted to obtain a renewal node key K(t)0, and an encrypted key Enc(K(t)0, K(t)R) in the first stage from the top of the EKB of FIG. 4A is decrypted to obtain K(t)R. On the other hand, in the devices ~~K 0000, K0001,~~ 0 and 1 a node key K000 is not included to be renewed~~,~~. ~~and a key necessary for a renewal node key is~~ The renewal keys are K(t)00, K(t)0~~,~~ and K(t)R. The devices 0 and 1 ~~K0000.K0001~~ decrypt~~s~~ an encrypted key Enc(K000, K(t)00) in the third stage from the top of the EKB of FIG. 4A to obtain K(t)00, and thereafter, an encrypted key Enc(K(t)00, K(t)0) in the second stage from the top of the EKB of FIG. 4A is decrypted, and an encrypted key Enc(K(t)0, K(t)R) in the first stage from the top of the EKB of FIG. 4A is decrypted to obtain K(t)R. By doing so, the devices 0, 1~~,~~ and 2 can obtain a renewed key K(t)R. The index in the EKB of FIG. 4A shows the absolute address of a node key and a leaf key used as a decryption key.

[0112] Where renewal of a node key : K(t)0, K(t)R in the upper stage in the tree structure shown in FIG. 3 is unnecessary, and ~~a~~ renewal processing of only the node key K00 is necessary~~,~~ ~~an~~ the enabling key block (EKB) shown in FIG. 4B can be used to distribute a renewal nod key K(t)00 to the devices 0, 1~~,~~ and 2.

[0113] The EKB shown in FIG. 4B can be used, for example, to distribute a new common content key ~~in common in~~ to a specific group. ~~Concretely~~Illustratively, it is supposed that the devices 0, 1, 2~~,~~ and 3 shown by the dotted line in FIG. 3 use a recording medium, and a new common content key K(t)con is necessary. At this time, Enc(K(t)00, K(t)con) ~~into which new common content key: K(t)con is encrypted with K(t)00 into which a common node key K00 of the devices 0, 1, 2 is renewed~~ is distributed with the EKB shown in FIG. 4B to devices 0, 1 and 2. By this distribution, distribution of data not decrypted in the apparatus of other groups such as a device 4 becomes enabled.

[0114] That is, if the devices 0, 1, and 2 decrypt the encrypted sentence using K(t)00 obtained by processing the EKB of Fig. 4B , a content key, K(t)con, at the time ~~t K(t)con~~ can be obtained.

~~[Distribution of a content key using EKB]~~

[0115] FIG. 5 shows~~, as~~ an example ~~of processing~~ for obtaining a content key, K(t)con , at the time t~~ K(t)con~~, ~~a processing of~~ in a device 0, which receives, through a recording medium, data Enc(K(t)00, K(t)con) (into which ~~a~~ the new common content key K(t)con is encrypted using K(t)00) and the EKB shown in FIG. 4B. That is, this is an example in which encrypted message data ~~by~~ in an EKB is a content key K(t)con.

[0116] As shown in FIG. 5, a device 0 uses ~~generation : EKB at generation: t stored in the recording medium and~~ a node key K000 stored in advance by itself to produce a renewal node key K(t)00 from the EKB by the EKB processing similar to that described above. Further, a renewal content key K(t)con is decrypted using ~~a~~ the renewal node key K(t)00 ~~decrypted,~~ and is encrypted by a leaf key K0000 owned by device 0 ~~itself~~ and then stored ~~in order to~~ for later use~~ it later~~.

~~[Format of EKB]~~

[0117] FIG. 6 shows an example of a format of the enabling key block (EKB). A version 601 is a discriminator showing the version

of the enabling key block (EKB). The version is ~~has a function~~ for use ~~showing a corresponding relation between a function for~~ in discriminating between the latest EKB and a content. The depth 602 ~~shows~~ provides the number of hierarchies of a hierarchical tree with respect to a device of the distributing destination of the enabling key block (EKB). A data pointer 603 is a pointer for indicating a position of data part ~~in~~ of the enabling key block (EKB), and a tag pointer 604 is a pointer for indicating a position of a tag ~~part~~part of the EKB, and a signature pointer 605 is a pointer for indicating a position of the signature part of the EKB.

[0118] ~~A d~~Data part 606 stores, for example, ~~data having a node key to be renewed encrypted. For example, it stores~~ various encrypted keys in connection with a renewal node key as shown in FIG. 5.

[0119] ~~A t~~Tag part 607 is a tag for indicating a positional relationship of encrypted node keys and leaf keys stored in the data part. An attaching rule of this tag will be described with reference to FIGS. 7A to 7C. FIGS. 7A to 7C show an example for sending the enabling key block (EKB) described previously in FIG. 4A as data. The data at that time is as shown in FIG. 7B. An address of a top node included in an encrypted key at that time is used as a top node address. In this case, since a renewal key of a root key K(t)R is included, a top node address is KR. At this time, for example, data Enc(K(t)0, K(t)R) in the uppermost stage is at a position shown in ~~a~~ the hierarchical tree shown in FIG. 7A. (The n~~N~~ext data is Enc(K(t)00, K(t)0), which is at a position under on the left hand of the previous data in the tree. Where data ~~is~~ exists, a tag is set to 0, and where data ~~is~~ does not exist, a tag is set to 1. The tag is set as (left (L) tag, right (R) tag). Here, s~~S~~ince data ~~is~~ exists at the left of the data at the top stage Enc(K(t)0, K(t)R), L tag = 0, and since data ~~is~~ does not exist to the right, R tag = 1. Tags are set to

all the data to constitute a row of data and a row of tags as, shown in FIG. 7C.

[0120] The tag is set in order to show at which position of the tree structure data Enc(Kxxx, Kyyy) is positioned. Since the key data Enc(Kxxx, Kyyy) ... are mere enumerated data of simply encrypted keys, a position on the tree of an encrypted key stored as data can be discriminated by the aforementioned tag. Alternatively, f~~F~~or example, data ~~constitution as in the following~~ as shown below can be provided using the node index placed in correspondence to the encrypted data ~~like the constitution described~~ as shown in FIGS. 4A and 4B previously without using the aforementioned tag:

1. 0: Enc(K(t)0, K(t)root)
2. 00: Enc(K(t)00, K(t)0)
3. 000: Enc(K(t)000, K(t)00)
4. ...

[0121] However, ~~the constitution~~ using such an index as ~~described~~ shown above results in a ~~lengthy data to increase data quantities,~~ larger size EKB, which is not preferable in ~~the~~ distribution through a network. On the other hand, use of the aforementioned tag ~~is used~~ as index data ~~showing~~ allows discrimination of a key position using ~~whereby a key position can be discriminated with~~ less data~~quantity~~.

[0122] Returning to FIG. 6, the EKB format will be further described. The signature is an electronic signature executed, for example, by a key control center, a content provider, a settlement organization or the like which issued the enabling key block (EKB). The device which received the EKB confirms, by authentication of the signature, that it is an enabling key block (EKB) issued by a valid enabling key block (EKB) issuer.

~~[Content Key Using EKB and Distribution of Contents]~~

[0123] While in the aforementioned example, ~~a description was made of an example in which~~ only the content key is sent along

with the EKB, a description will be made hereinafter ~~of the constitution~~ in which encrypted ~~a~~ content ~~encrypted~~ is also sent~~by a content key, and a content key encrypted by a content encrypted key along with a content key encryption key encrypted by EKB are sent~~.

[0124] This is shown in FIGS. 8A and 8B~~ show this data constitution~~. In ~~the constitution shown in~~ FIG. 8A, Enc(Kcon, content) 801 is data in which ~~a~~ content is encrypted by a content key(Kcon), Enc(KEK, Kcon) 802 is data in which a content key (Kcon) is encrypted by a content key-encryption key (KEK)~~: Key Encryption key)~~, and Enc(EKB, KEK) 803 is data in which a content ~~key-encryption key~~ KEK is encrypted by an enabling key block (EKB).

[0125] Here, the content key-encryption key (KEK) may be a node key (K000, K00 ...) or a root key (KR) itself, and may be a key encrypted by a node key (K000, K00 ...) or a root key (KR).

[0126] FIG. 8B shows an example ~~of constitution~~ where a plurality of contents are recorded in media, which makes use of the same Enc(EKB, KEX) 805. In such a case~~constitution as described~~, the same Enc(EKB, KEK) is not added to each data, but data showing a link ~~linking destination linked~~ to Enc(EKB, KEK) is added to each data.

[0127] FIG. 9 shows an example ~~of a case~~ where a content encryption key KEK is ~~constituted as~~ a renewal node key K(t)00 obtained by renewal of ~~ed~~ the node key K00 shown in FIG. 3. In this case, if in a group surrounded by the dotted line ~~frame~~ in FIG. 3, the device 3 is revoked, for example, due to the leak of a key, data having an enabling key bock (EKB) shown in FIG. 9 and data into which a content key (Kcon) is encrypted by a content key encryption key (KEK = K(t)00), and data into which a content is encrypted by a content key (Kcon) are distributed to members of the other groups, that is, devices 0, 1, 2 whereby the devices 0, 1~~,~~ and 2 can obtain the content.

[0128] The right side in FIG. 9 shows the decrypting procedure in the device 0. The device 0, first, obtains a content key encryption key (KEK = K(t)00) from the received EKR by performing a decrypting process using a leaf key K000 held by itself ~~from the received enabling key bock~~. Then, the device 0 obtains a content key Kcon decrypted by the key K(t)00, and further carries out decrypting by the content key Kcon. The device 0 can use the content as a result of the above process. The devices 1, 2 are also able to obtain a content key encryption key (KEK=K(t)00) by processing the EKB in a similar fashion ~~by the different procedures~~ and are able to use the content similarly.

[0129] The devices 4, 5, 6 ... of the other groups shown in FIG. 3 are not able to obtain a content key encryption key (KEK = K(t)00) using a leaf key and a node key held by themselves even if they receive the same ~~data (~~EKB~~)~~ as mentioned above. The revoked device 3 ~~revoked~~ is likewise not able to obtain the content key encryption key (KEK = K(t)00) by a leaf key and a node key, and only the device having the proper right is able to decrypt and use the content.

[0130] If the distribution of a content key making use of the EKB is used, in a manner as described, the encrypted content ~~which only valid right holder can decrypt~~ can be distributed safely to only valid users.

[0131] An enabling key block (EKB), a content key, an encrypted content or the like can be safely distributed ~~has a constitution capable of providing distribution safely~~ through a network, but the enabling key block (EKB), the content key and the encrypted content can ~~be~~ also be stored in a recording medium such as DVD, CD and provided to a user. In this case, ~~if constitution is made such that a content key obtained by decrypting an enabling key block (EKB) stored in one and the same recording medium is used for decrypting the encrypted content stored in the recording medium, distribution process of an encrypted content that can be~~

~~used only with a leaf key and a node key held in advance by the valid right holder only, that is,~~ content distribution can be further ~~for which a usable user's device is~~ limited ~~can be realized~~ by a simple structure~~constitution~~.

[0132] FIG. 10 shows an example of constitution in which an enabling key block (EKB) is stored together with an encrypted content ~~are stored~~ in a recording medium. In the example shown in FIG. 10, stored in the recording medium are contents C1 to C4, data associating an ~~with the~~ enabling key block corresponding to each stored content ~~placed in correspondence thereto~~, and an enabling key block of version M (EKB_M). For example, EKB_1 is used to produce a content key Kcon1 having a content C1 encrypted, and for example, EKB_2 is used to produce a content key Kcon2 having a content C2 encrypted. In this example, an enabling key ~~bock~~ block of version M (EKB_M) is stored in a recording medium. Since contents C3, C4 are ~~is~~ placed in correspondence to the enabling key block (EKB_M), ~~contents of~~ the contents C3, C4 can be obtained by decrypting the enabling key block (EKB_M). Since EKB_1, EKB_2 are not stored in the recording medium~~a disk~~, it is necessary to obtain EKB_1, EKB_2 ~~necessary for decrypts the respective content keys~~ by new distribution means, for example, network distribution or distribution by a recording medium.

[0133] FIGS. 11A and 11B show a comparative example between a content key distribution ~~by~~ using EKB and conventional content key distribution where a content key is circulated among a plurality of devices. FIG. 11A shows the conventional approach~~constitution~~, and FIG. 11B shows an example making use of an enabling key block (EKB) according to the present invention. In FIGS. 11A and 11B, Ka (Kb) indicates data in which Kb is encrypted by Ka.

[0134] As shown in FIG. 11A, processing has been heretofore carried out in which validity of a data transmit-receiver is

confirmed, authentication processing and authentication and key exchange (AKE) are executed between devices to co-own a session key, Kses, ~~used in encrypting process of data transmission,~~ and a content key Kcon is encrypted by the session key, Kses, under the condition that the authentication is established to effect transmission.

[0135] For example, in the PC shown in FIG. 11A, it is possible to decrypt a content key, Kcon, ~~Kses~~ encrypted by the ~~a~~ session key, Kses ~~received by the session key to obtain Kcon~~, and further possible to encrypt Kcon ~~obtained~~ by a stored key, Kstr, held by the PC itself to store, Kstr (Kcon) ~~it~~ in its own memory.

[0136] In FIG. 11A, ~~processing is necessary in which even where data is desired to be distributed in the form capable of being used for only a recording device 1101 shown in FIG. 11A, when PC or a reproducing device is present,~~ authentication processing as shown in FIG. 11A is executed so that content keys are encrypted by the respective session keys to effect distribution even where data is desired to be distributed in the form capable of being used for only a recording device 1101 shown in FIG. 11A. The PC or the reproducing device is likewise able to use a session key produced in the authentication process and co-owned to decrypt an encrypted content key ~~and obtain a content key~~.

[0137] On the other hand, in an example making use of an enabling key block (EKB) shown in ~~the lower stage of~~ FIG. 11B, an enabling key block (EKB), and data (Kroot (Kcon)) having a content key Kcon encrypted by a node key or a root key obtained by processing the enabling key block (EKB) are distributed from a content provider, whereby the content key Kcon can be decrypted and obtained ~~by~~ only by the apparatus capable of processing the distributed EKB ~~distributed~~.

[0138] Accordingly, for example, the useable enabling key block (EKB) is produced only on the right end in FIG. 11B, and the enabling key block (EKB), and data having an encrypted content

key Kcon ~~encrypted by a node key or a root key obtained by EKB processing~~ are sent together whereby the PC, the reproducing apparatus or the like present cannot execute processing of the EKB by a leaf key or node key owned by itself. Accordingly, the useable content key can be distributed to only a ~~the~~ valid device ~~safely~~ without executing processes such as authentication ~~process between the data transmit-receive devices~~, the production of a session key, and the process for encrypting a content key Kcon by the session key as illustrated in FIG. 11A.

[0139] Where the useable content key is desired to be distributed to PC, a recording and reproducing unit also, an enabling key block (EKB) capable of being processed is produced and distributed to thereby obtain a common content key.

~~[Distribution Of Authentication Key Using Enabling Key Block (EKB) (Common Key System)]~~

[0140] In the distribution of data used in the enabling key block (EKB) or a key described above, since an enabling key block (EKB) and a content or a content key which are transferred between devices always maintain the same encryption form, there is the possibility that an invalid copy is produced due to the so-called replay attack, which ~~steals and~~ records a data transmission channel and transfers it ~~later~~ again later. For preventing such an attack as described, there is an effective means for executing an authentication ~~process~~ and key exchange process similar to those of the prior art between data transfer devices. Now, a description is made of an arrangement ~~the constitution~~ in which an authentication key, Kake, used when the authentication process and key exchange process are executed, is distributed to a device using the aforementioned enabling key block (EKB), whereby the authentication process is in conformity with a common key system having a common authentication key as a safe private key ~~is executed~~. That is, this is an example in which encrypted message data of the ~~by~~ EKB is used as an

authentication key.

[0141] FIG. 12 shows a mutual authentication method (ISO/IEC 9798-2) using a common key encryption system. While in FIG. 12, DES is used as the common key encryption system, other systems may be used as long as they are the common key encryption system. In FIG. 12, first, B produces the random number Rb of 64 bits, and Rb and ID (b), which is its own ID, are transmitted to A. A, which receives them, newly produces the random number Ra of 64 bits, and data (Ra, Rb, ID(b)) are encrypted using a key Kab in the CBC mode of DES ~~in order to Ra, Rb~~ and ~~Rc to~~ transmitted ~~them~~ to B. The key Kab is a key to be stored in a recording element as a private key common to A and B. According to the encrypting processing by the key Kab using the CBC mode of DES, for example, ~~in the processing using DES~~, an initial value and Ra are subjected to an exclusive OR; in the DES encryption part, the key Kab is used for encrypting to generate an encrypted text E1~~and continuously~~.~~,~~ ~~t~~The encrypted text E1 and Rb are subjected to an exclusive OR; in the DES encryption part, a key Kab is used for encrypting~~,~~ to generate ~~and~~ encrypted text E2. The encrypted text E2 and ID (b) are subjected to an exclusive OR; and in the DES encryption part, a key Kab is used for encrypting to generate encrypted text ~~transmission data~~ (Token-AB) ~~by an encrypted text E3 produced~~. The token-AB [E1, E2, E3] is transmitted to B.

[0142] B~~, which received the above data,~~ decrypts the received token-AB, ~~data by~~ a key Kab (authentication key) likewise stored in a recording element as a common private key. ~~A decrypting method of received data, f~~First, B decrypts ~~an~~ encrypted text E1 by ~~an~~ authentication key Kab to obtain the random number Ra. Next, ~~an~~ encrypted text E2 is decrypted by ~~an~~ authentication key Kab, and the result therefrom and E1 are subjected to exclusive OR to obtain Rb. Finally, ~~an~~ encrypted text E3 is decrypted by an authentication key Kab, and the result therefrom and E2 are subjected to exclusive OR to obtain ID (b). ~~Authentication is~~

~~made~~ B authenticates that A is valid if Ra and ID (b) out of Ra, Rb and ID (b) thus obtained are coincided with the ones transmitted by B. ~~When passed this authentication, B authenticates that A is valid.~~

[0143] Next, B produces a session key (Kses) to be used after authentication (Producing method: To use the random number). Then, Rb, Ra~~,~~ and Kses are encrypted in that order using an authentication key Kab in the CBC mode of DES and are returned to A.

[0144] A, which received the above data, decrypts the received data by ~~an~~ authentication key Kab. A decrypting method of the received data is similar to the decrypting process of B, which is therefore omitted in its detail. ~~Authentication is made~~ A authenticates that B is valid if Rb and Ra out of Rb, Ra and Kses thus obtained are coincided with the ones transmitted by A. When passed the authentication~~, A authenticates that B is valid~~. After authentication ~~of mating parties each other~~, the session key, Kses, is used as a common key for secrete communication after authentication.

[0145] Where invalidity ~~or uncoincidence~~ is found when the received data are authenticated, processing is interrupted as a failure of mutual authentication.

[0146] In the above-described authentication process, A and B co-own a common authentication key Kab. The common authentication key Kab is distributed to a device using the enabling block key (EKB).

[0147] For example, with reference to ~~in the example shown in~~ FIG. 12, there may be employed the arrangement ~~constitution~~ in which out of A or B, the other encrypts an authentication key Kab by ~~and~~ an enabling key block (EKB) ~~produced by producing a decodable enabling key block (EKB)~~ to transmit it to the other, or the arrangement ~~constitution~~ in which a third party produces an enabling key bock (EKB) that can be used by both devices A and

B for the devices A and B to encrypt an authentication key Kab by the enabling key block (EKB) ~~produced for the devices A, B~~ to distribute it.

[0148] FIGS. 13 and 14 show examples ~~of the constitution~~ in which an authentication key, Kake, common to a plurality of devices is distributed by an enabling key block (EKB). FIG. 13 shows an example in which a decodable authentication key, Kake, is distributed to devices 0, 1, 2~~,~~ and 3, and FIG. 14 shows an example in which the device 3 out of the devices 0, 1, 2~~,~~ and 3 is revoked to distribute a decodable authentication key to only the devices 0, 1~~,~~ and 2.

[0149] In the example of FIG. 13, a node key K(t)00 is renewed using a node key and a leaf key in the devices 0, 1, 2, 3 ~~is produced and distributed,~~ by producing a decodable enabling key block (EKB), along with data (b) having an authentication key Kaka decrypted by the ~~a~~ renewed ~~al~~ node key K(t)00. First, the respective devices, as shown on the right side of FIG. 13, processes (decrypts) EKB to thereby obtain a renewed node key K(t)00, and then decrypts an authentication key: Enc(K(t)00, Kake) encrypted using the obtained node key K(t)00 to obtain the ~~an~~ authentication key Kake.

[0150] In the other devices 4, 5, 6, 7 ..., even if the same enabling key block (EKB) is received, the node key K(t)00 renewed by processing EKB cannot be obtained, and therefore, an authentication key can be sent to only the valid device safely.

[0151] On the other hand, ~~the example of~~ FIG. 14 shows ~~is~~ an example in which ~~as~~ the device 3 is~~, for example,~~ revoked. ~~by leak of a key, the device 3 in a group surrounded by the dotted frame of FIG. 3 produces a~~A decodable enabling key block (EKB) is produced with respect to the only other members of the ~~other~~ group, that is, the devices 0, 1~~,~~ and 2 for distribution. Data having (a) an enabling key block (EKB) and (b) an authentication key (Kake) ~~shown in FIG. 14~~ (encrypted by the node key (K(t)00))

are distributed.

[0152] On the right side of FIG. 14, the decrypting procedure is shown. First, the devices 0, 1~~,~~ and 2 obtain~~s~~ an enabling node key (K(t)00) by performing a decrypting process using a leaf key or a node key owned by itself from the received enabling key block. Next, the devices obtain the ~~an~~ authentication Key Kake by decrypting Enc(k(t)00,Kake) ~~made by K(t)00~~.

[0153] The devices 4, 5, 6 ... in the other group shown in FIG. 3 cannot obtain a renewal node key (K(t)00) using a leaf key and a node key owned by itself even if similar data (EKB) is received. Similarly, ~~also~~ in the revoked device 3 ~~revoked~~, the renewal node key (K(t)00) cannot be obtained by a leaf key and a node key owned by itself. ~~, and~~Thus, only the device having a valid right is able to decrypt an authentication key for use.

[0154] If distribution of an authentication key making use of an EKB is used, only the valid right holder is able to distribute a decodable authentication key safely with less data quantity.

~~[Distribution of content key using a public key authentication and an enabling key block (EKB)]~~

[0155] In the following, the distribution process of the content key using a public key authentication and an enabling key block (EKB) will be described. First, a mutual authentication method using an elliptic curve encryption of 160-bit length, which is a public key encryption system, will be described with reference to FIG. 15. In FIG. 15, ECC is used as the public key encryption system, but any system may be used as long as it is a public key encryption system similar thereto. Further, the key size need not be 160 bits. In FIG. 15, first, B produces the random number Rb of 64 bits to transmit it to A. A, which received it, newly produces the random number Ra of 64 bits, ~~and~~ the random number Ak smaller than the prime number p, and a ~~. And, a~~ point Av = Ak x G is obtained (Av is 160 bit). ~~by making a base point G, Ak times is obtained to produce a~~An electronic signature A.Sig is

produced ~~A, Sig~~ with respect to Ra, Rb, Av (X coordinate and Y coordinate, each 64 bits), which is returned, along with a public certificate of A, to B. ~~In Ra and Rb, X coordinate and Y coordinate of 64 bits, Av are respectively 160 bits, and therefore, a~~An electronic signature comprising up to ~~with respect to~~ 448 bits in total is produced.

[0156] B, which received the public key certificate, Ra, Rb, Av, and the electronic signature A.—Sig, authenticates if Rb transmitted by A is the same as the ~~coincided with~~ one produced by B. As a result, when they are the same~~coincided~~, an electronic signature within the public key certificate of A is authenticated by a public key of an authentication office to take out a public key of A. The electronic signature A.—Sig is authenticated using the ~~a~~ public key of A ~~taken out~~.

[0157] Next, B produces the random number Bk which is smaller than the prime number p. A point Bv = Bk x G is obtained ~~by making a base point G Bk times is obtained~~ to produce an electronic signature B.—Sig with respect to Rb, Ra, Bv (X coordinate and Y coordinate), which is returned to A along with a public key certificate of B.

[0158] A, which received the public key certificate, Rb, Ra, Av, and the electronic signature B.—Sig of B authenticates if Ra transmitted by B is coincided with the one produced by A. As a result, when they are the same~~coincided~~, an electronic signature within the public key certificate of B is authenticated by a public key of an authentication office to take out a public key of B. The electronic signature B. Sig is authenticated using the ~~a~~ public key of B ~~taken out~~. After the authentication of an electronic signature has been succeeded, A authenticates B to be valid.

[0159] Where both of them have succeeded in ~~for~~ authentication, B computes Bk x Av (~~S~~since Bk is the random number, but Av is the point on the elliptic curve, scalar-times computation at the

point on the oval curve is necessary~~.~~), and A computes Ak x Bv, and uses the lower 64 bits of the X coordinate of these points as a session key for use ~~for~~ thereafter ~~communication~~ (where a common key encryption is ~~a common key encryption~~ of 64 bit key length). Of course, a session key may be produced from the Y coordinate, and the coordinate need not be the lower 64 bits. ~~In~~Something in the secrete communication after mutual authentication~~, sometimes,~~ the transmission data is not only encrypted by a session key but is also applied with an electronic signature.

[0160] Where in the authentication of an electronic signature or authentication of the received data, invalidity ~~or uncoincidence~~ is found, processing is interrupted due to a failure of mutual authentication.

[0161] FIG. 16 shows an example of a distribution process of content keys using a public key authentication and an enabling key block(EKB).~~,~~ First, the authentication process according to the public key system described above ~~explained referring to FIG. 15~~ is executed between a content provider and a PC. The content provider produces a decodable EKB comprising a renewed node key and a content key encrypted with the renewable key (E(Kcon)). ~~by a reproducing apparatus which is a content key distribution destination, a node key and a leaf key owned by a recording medium to encrypt a content key E(Kcon) which executed encryption by a renewal node key and an enabling key block (EKB) by a session key Kses produced by the authentication process between PCs, which is~~ In addition, the EKB and E(Kcon) are encrypted using the session key Kses and transmitted to the PC.

[0162] The PC decrypts the received data using the session key, Kses ~~[a content key E (Kcon)] which executed encryption by a renewal node key and an enabling key block (EKB)] encrypted by a session key,~~ and thereafter transmits it to a reproducing apparatus and a recording medium.

[0163] The reproducing apparatus and the recording medium receives the renewed key from the EKB as described earlier to further recover the content key, Kcon. ~~decrypt [a content key E (Kcon) which executed encryption by a renewal node key and an enabling key block (EKB)] to thereby obtain a content key Kcon.~~

[0164] According to the above arrangement~~constitution~~, since encrypted data using an EKB ~~[a content key E (Kcon) which executed an encryption by a renewal node key and an enabling key block (EKB)]~~ are transmitted under the condition of the authentication between a content provider and PC, for example, even in the case where a node key is leaked, positive data transmission to a mating party is enabled.

~~[Distribution of a program code by using an enabling key block (EKB)]~~

[0165] While in the above-described example, a description has been made of a method for encrypting a content key, an authentication key or the like using an enabling key block (EKB) to distribute it, an arrangement ~~the constitution~~ in which various program codes are distributed using an enabling key block (EKB) may be employed. That is, this is an example in which encrypted message data of an ~~by~~ EKB is used as a program code. ~~This constitution will be described hereinafter.~~

[0166] FIG. 17 shows an example in which a program code is encrypted~~, for example,~~ by a renewal node key of an enabling key block (EKB) to transmit it between devices. A device 1701 transmits, to device 1702 an enabling key block (EKB) that can be decrypted by a node key and a leaf key of a device 1702, and a program code subjected to decrypting by a renewal node key contained in the enabling key block (EKB) ~~to a device1702~~. The device 1702 processes the received EKB to obtain the ~~a~~ renewal node key, and further executes decrypting of the ~~a~~ program code by the obtained ~~a~~ renewal node key ~~obtained to obtain a program code~~.

[0167] In the example shown in FIG. 17, further, processing by the program code obtained in the device 1702 is executed to return the result to the device 1701, and the device 1701 further continues processing on the basis of the result.

[0168] As described above, the enabling key block (EKB) and the program code (subjected to decrypting processing by the renewal node key contained in the enabling key block (EKB)) are distributed whereby a program code capable of being decrypted in a specific device can be distributed to the specific device or the group shown in FIG. 3.

~~[Constitution for causing ICV: Integrity Check Value to correspond to a transmission content]~~

[0169] Next, a description will be made of the processing arrangement in which ~~constitution in which for preventing falsification of a content,~~ an ~~the~~ integrity check value (ICV) is produced to correspond to the content. ~~, and t~~The presence or absence of the falsification of the content is judged by using the ~~computing~~ ICV.

[0170] The integrity check value (ICV) is, for example, computed using a hash function with respect to the content, and is computed by ICV =hash (Kicv, C1, C2, ...) . Kicv is an ICV producing key. C1, C2 are content information. ~~information of a content, and a~~A message authentication code (MAC) of content ~~important~~ information ~~of the content~~ is also used.

[0171] FIG. 18 shows an example for producing a MAC value ~~producing example~~ using the DES encryption processing arrangement~~constitution~~. As shown in ~~the constitution of~~ FIG. 18, a message ~~to be an object~~ is divided into 8-bit units (hereinafter, the divided messages are M1, M2, … MN). First, the initial value (hereinafter, IV) and M1 are subjected to exclusive OR (the result of which is I1). Next, I1 is put into a DES encryption part to carry out encrypting using a key (hereinafter, K1) (the ~~an~~ output is E1). Continuously, E1 and M2 are subjected

to exclusive OR, the output ~~I2~~ of which, I2, is put into the DES encryption part–, and is encrypted using the key 1 (the ~~an~~ output E2). Thereafter, this procedure is repeated, and the encrypting processing applied to all of the messages. The last EN is the ~~a~~ message authentication code (MAC).

[0172] The hash function is applied to the MAC value of the content and the ICV producing key to produce the integrity check value (ICV) of the content. An ICV produced for ~~when a~~ content ~~is produced~~ for which ~~the fact that~~ no falsification ~~is present~~ is assured is compared with an ICV produced on the basis of ~~a~~ new content. If the same ICV is obtained, the fact that the content is not falsified is assured, and if the ICVs are ~~is~~ different, a judgment that falsification is present can be ~~is~~ made.

~~[Constitution for distributing a producing key Kicv of the check value (ICV) by EKB]~~

[0173] Next, an arrangement ~~the constitution~~ in which the Kicv ~~Kicc which is an integrity check value (ICV) producing key of a content~~ is sent by the enabling key block will be described. That is, this is an example in which encrypted message data of an ~~by~~ EKB is an integrity check value (ICV) producing key ~~of a content~~.

[0174] FIG. 19 and FIG. 20 show ~~an~~ examples in which (where contents common to a plurality of devices are sent)~~,~~ an integrity check value producing key Kicv ~~for authenticating the presence or absence of falsification of these contents~~ is distributed by the enabling key block (EKB). FIG. 19 shows an example in which the ~~a decodable integrity check value producing key~~ Kicv is distributed to devices 0, 1, 2~~,~~ and 3, and FIG. 20 shows an example in which the device 3 ~~out of the devices 0, 1, 2, 3~~ is revoked, and the ~~a decodable integrity check value producing key~~ Kicv is distributed to only the devices 0, 1~~,~~ and 2.

[0175] In the example of FIG. 19, a node key K(t)00 (renewed using a node key and a leaf key owned by the devices 0, 1, 2~~,~~ and

3) along with data (b) having a ~~check value producing key~~ Kicv encrypted by the ~~a~~ renewed ~~al~~ node key K(t)00 are distributed by producing a decodable enabling key block (EKB). As shown on the right side in FIG. 19, the respective devices first process (decrypt~~s~~) the EKB to thereby obtain the renewed ~~a~~ node key K(t)00 ~~renewed~~, and subsequently decrypt a check value producing key : Enc(K(t)00, Kicv) encrypted using the obtained renewed node key K(t)00 to obtain the ~~a~~ check value producing key Kicv.

[0176] Since other devices 4, 5, 6, 7 ... cannot obtain the renewed ~~a~~ node key K(t)00 ~~renewed~~ by processing the EKB by a node key and a leaf key owned by themselves ~~itself~~ even if the same enabling key block (EKB) is received, the check value producing key, Kicv, can be safely sent to only valid devices ~~safely~~.

[0177] On the other hand, ~~the example of~~ FIG. 20 is an example in which as a ~~the~~ device 3 is, for example, revoked by leak of a key, in a group surrounded by the dotted frame of FIG. 3. ~~produces a~~ A decodable enabling key block (EKB) is produced for distribution, with respect to the only other members of the ~~other~~ group, that is, the devices 0, 1~~,~~ and 2 ~~for distribution~~. Data having (a) an enabling key block (EKB) and (b) a check value producing key (Kicv) shown in FIG. 20 encrypted by the renewed node key (K(t)00) are distributed.

[0178] On the right side of FIG. 20, the decrypting procedure is shown. First, the devices 0, 1~~,~~ and 2 obtain the renewed ~~a renewal~~ node key (K(t)00) by performing a decrypting process using a leaf key or a node key owned by itself from the received enabling key block. Next, the devices obtain a check value producing key, Kicv, by decrypting Enc (K(t)00, Kicv) ~~made by K(t)00~~.

[0179] The devices 4, 5, 6 ... ~~in the other~~ outside the group shown in FIG. 3 cannot obtain ~~a~~ the ~~renewal~~ renewed node key (K(t)00) using a leaf key and a node key owned by ~~itself~~ themselves even if similar data (EKB) is received. Similarly,

also in the revoked device 3~~ revoked~~, the ~~renewal~~ renewed node key (K(t)00) cannot be obtained by a leaf key and a node key owned by itself. ~~, and o~~Only ~~the~~ a device having a valid right is able to decrypt an authentication key for use.

[0180] If distribution of a check value reproducing key making use of an EKB is used, only ~~the~~ a valid right holder is able to distribute a decodable check value producing key safely, and with less data ~~quantity~~ overhead.

[0181] By using the integrity check value (ICV) of contents as described above, it is possible to eliminate invalid copies of an EKB and encrypted contents. ~~It is supposed that f~~For example, as shown in FIGS. 21A and 21B, there is a medium .1 in which a content C1 and a content C2 are stored along with an enabling key block (EKB) that is capable of ~~obtaining~~ providing content keys. The content C1 and C2 along with the associated EKB, are ~~, which is~~ copied to a medium 2 without modification. ~~It is possible to copy EKB and encrypted contents, which~~The copied content can be used in a device capable of decrypting the associated EKBs.

[0182] ~~There is provided a constitution~~However, in Fig. 21B there is provided an arrangement in which ~~as shown in FIG. 21B,~~ integrity check values (ICV (C1, C2)) are also stored corresponding to stored contents. ~~properly stored in the respective media.~~ The notation (ICV (C1, C2)) ~~shows~~ is representative of ICV = hash (Kicv, C1, C2) in which ~~is~~ an integrity check value ~~of contents~~ is computed using the hash function ~~in~~ on the content C1 and the content C2. ~~In the constitution of~~ As shown in FIG. 21B, a content 1 and a content 2 are ~~properly~~ stored in the medium 1, and integrity check values (ICV (C1, C2)) produced on the basis of the content C1 and the content C2 are stored. Further, a content 1 is properly stored in the medium 2, and an integrity check values (ICV (C1)) produced on the basis of the content C1 is stored therein. In this ~~constitution, Assume~~ example it is assumed, that (EKB, content

2) stored in the medium 1 is to be ~~is~~ copied to the medium 2. ~~, when in the medium 2,~~In this process a content check value is newly produced, ICV (C1, C2). ~~are to be produced, so that it becomes obvious that~~This is obviously different from the value of Kicv (C1) already stored in the medium 2. ~~, falsifying of contents and storing of new contents due to the invalid copy are executed.~~ In the reproducing device ~~for reproducing~~ media, ICV checking is executed prior to actually copying (EKB, content 2) to medium 2 ~~in the step previous to the reproducing step,~~ and a judgment is made ~~of coincidence between~~ if the produced ICV and the stored ICV are the same. ~~, if not coincident, the constitution in which reproducing is not executed is provided to enable prevention of reproducing contents copied invalidly.~~In this example, the ICVS are not the same and no copying occurs. If the ICVS had been identical, the copying would be permitted.

[0183] Furthermore, there can be provided an arrangement ~~the constitution in which~~ for enhancing safety, in which the integrity check value (ICV) of the contents is ~~rewritten to produce them~~ produced on the basis of data including a counter value. That is, ~~this constitution is to make computation by~~ ICV = hash (Kicv, counter + 1, C1, C2, ...). Here, a counter (counter + 1) is ~~set as a value in which one increment is made every rewriting.~~incremented for every rewrite. It is necessary to store ~~have a constitution in which a~~ the counter value ~~is stored~~ in a secure memory.

[0184] Further, in ~~the constitution~~an arrangement, in which the integrity check value (ICV) of the contents ~~is~~ cannot be stored in the same medium as the contents, the integrity check value (ICV) of the contents is stored in a separate medium.

[0185] For example, where contents are stored in media for which ~~take~~ no measures are taken to prevent copies (such as a read only memory or normal MO), there is the possibility that when the integrity check value (ICV) is stored in the same medium,

rewriting of the ICV is done by an invalid user, thus failing to safely maintain the ~~safety of~~ original ICV. In such a case, there can be provided ~~the constitution~~an arrangement in which an ICV is safely stored in a ~~safety~~ medium on a host machine, and the ICV is used for copy control (for example, check-in / check-out, move), to thereby enable ~~safe~~ management of the ICV and checking for ~~of~~ falsification of contents.

[0186] The above ~~constitution~~ arrangement is shown in FIG. 22. In FIG. 22, contents are stored in a medium 2201, which takes no measures for preventing copying such as read only media or normal MO. ~~,; and t~~The integrity check values (ICV) in connection with these contents are stored in a safe media 2202 on a host machine to which a user is not allowed to get access to prevent invalid rewriting of the integrity check value (ICV) by ~~a~~ the user. If, ~~as such a constitution as described above, for example, employment is made of a constitution in which when~~ a device on which ~~a~~ media 2201 is mounted executes reproducing of the media 2201, a PC or a server, which is a host machine, ~~executes checking of~~check the ICV to judge the propriety of reproducing. ~~,~~ Thus, reproducing of an invalid copy ~~contents or falsified contents~~ can be prevented.

~~[Category classification of a hierarchical tree structure]~~

[0187] ~~A description has been made of the constitution in which an encrypted key is constituted as a hierarchical tree structure shown in FIG. 3 such as a root key, a node key, a leaf key, etc., and~~ As described above, encrypted data (e.g., a content key, an authentication key, an ICV producing key or a program code, data or the like) are encrypted along with an enabling key block and are distributed. The EKB comprise keys representing node keys and leaf keys of a hierarchical tree structure as shown in FIG. 3. Now a ~~, but~~ a description will be made ~~hereinafter~~ of an arrangement ~~the constitution~~ in which the node and leaves of a

hierarchical tree structure are associated with categories. ~~which defines a node key or the like is classified every category of devices to execute efficient key renewing process, encrypted key distribution, and data distribution.~~

[0188] FIG. 23 shows one example of a category classification scheme for ~~classification of category of~~ a hierarchical tree structure. In FIG. 23, a root key Kroot 2301 is set on the uppermost stage of the hierarchical tree structure, a node key 2302 is set in the intermediate stage, and a leaf key 2303 is set in the lowest stage. Each device holds a respective individual leaf key~~s~~, and a series of node keys from the ~~a~~ leaf key to a root key, and the ~~a~~ root key.

[0189] ~~Here, as one example, nodes from the uppermost stage to the M stage is set as a category node 2304. That is,~~ In this example, each of nodes on the M stage is set as a device setting node of a specific category. Nodes and leaves lower than the M+1 stage are taken as nodes and leaves in connection with devices contained in the category thereof with one node in the M stage as a top.

[0190] For example, a category [Memory stick (trademark)] is set to ~~one~~ node 2305 in the M stage of FIG. 23. As a result, ~~and~~ nodes and leaves ~~provided~~ lower than ~~the~~ node 2305 are now set as ~~category-exclusive use~~ nodes or leaves containing various devices using the memory stick. ~~That is, those below the node 2305 are defined as the gathering of nodes and leaves associated with device defined in the category of the memory stick.~~

[0191] Further, a stage at a level below several stages from the M stage can be set as a sub-category. ~~node 2306.~~ For example, node 2306 is set as a node of [Reproducing exclusive-use unit], ~~is set as~~ a sub-category node contained in the category of the device using the memory stick. Node 2306 is ~~in a node~~ two stages below the ~~a~~ category [memory stick] ~~node 2305~~ as shown in the figure. Further, a node 2307~~of~~ associated with a telephone with a

music reproducing function would now be contained in the category associated with node 2306 ~~of~~ (the reproducing exclusive-use unit) ~~below the node 2306 of the reproducing exclusive-use unit~~ as a sub-category node. Similarly, ~~and~~ a [PHS] node 2308 and a [Portable telephone] node 2309 under node 2307 would now be contained in the category of the telephone with a music reproducing function ~~can be set therebelow~~.

[0192] Further, the category and sub-categories can be set not only with ~~at~~ the kind of devices, but also represents device independent categories. ~~at nodes managed independently, f~~For example, as makers, a content provider, a settlement organization or the like, ~~that is, at suitable units such as processing unit, jurisdiction unit, or service providing unit~~ (these will be generally called entity). For example, if one category node is set as a game machine XYZ exclusive-use top node (sold by game machine makers), a node key and a leaf key in the lower stage below the top node can be stored in any actual sold ~~the~~ game machine XYZ ~~sold by makers for sales~~. ~~, a~~After which, distribution of encrypted contents, or distribution of various keys, and renewal processing are distributed through ~~producing~~ an enabling key block (EKB) comprising ~~constituted by~~ node keys and leaf keys below the top node key. Thus, ~~and~~ data can be distributed only ~~that can be utilized merely~~ for use by the devices below the top node ~~can be distributed~~.

[0193] An arrangement ~~The constitution~~ can also be provided in which the node below a set top ~~one~~ node ~~as a top~~ is defined ~~set~~ as an associated node of the category or sub-categories defined, whereby makers, a content provider or the controlling ~~one~~ top node in the category stage or sub-category stage independently produce~~s~~ an enabling key block. The EKB can be distributed ~~with the node as a top to distribute it~~ to the devices belonging to those below the top node, and key renewal ~~ing~~ can be executed without affecting ~~at all on the~~ devices belonging to ~~the~~ nodes of

other categories not belonging to the top node.

~~[Key distributing constitution by simplified EKB (1)]~~

[0194] ~~For example, in the tree structure of FIG. 3 described previously, where for example, a content key is addressed to a predetermined device (leaf), a decodable enabling key block (EKB) is produced and provided using a leaf key and a node key owned by a key distributing device.~~ For example, in the ~~a~~ tree structure shown in FIG. 24A, ~~where~~ a key, for example, a content key, is to be transmitted to devices a, g, j [associated with leaf nodes Ka, Kg and Kj] ~~constituting a leaf~~. In this regard, a decodable enabling key block (EKB) is produced in the nodes Ka, Kg and Kj ~~a, g, j~~ and distributed.

[0195] It is also contemplated that, for example, a content key, K(t)con, is subjected to encrypting processing by a renewal root key, K(t)root, to distribute it along with EKB. In this case, the devices a, g, j execute processing to decrypt the received ~~of~~ EKB using a leaf key and a node key shown in FIG. 24B to obtain the renewed K(t)root. Once the latter is obtained, each device decrypts Enc (K(t)00, K(t)con to obtain the ~~, and execute decrypting process of a content key K(t)con by the obtained renewal root key K(t)root to obtain a~~ content key.

[0196] The arrangement ~~constitution~~ of the enabling key block (EKB) ~~(ERK)~~ provided in this case is as shown in FIG. 25. The format of the enabling key block EKB ~~(ERK)~~ shown in FIG. 25 is ~~constituted~~ in accordance with the format of the enabling key~~.~~ block (EKB) explained previously with reference to FIG. 6~~, has a tag corresponding to data (encrypted key).The tag is 0, if data is present in the directions of left (L) and right (R), and is 1 if not, as previously explained with reference to FIGS. 7A to 7C~~.

[0197] As described before, a ~~The~~ device which receives~~d~~ the enabling key block (EKB) sequentially executes decrypting process of the encrypted keys on the basis of an encrypted key of the enabling key block (EKB) and the tag to obtain a renewal key of

an upper node. As can be observed from ~~shown in~~ FIG. 25, in the enabling key block (EKB), the more the number of stages (depth) from a root to a leaf of a tree, the larger the ~~quantity of~~ depth~~s increases~~. In addition, ~~T~~the number of stages (depth) increases according to the number of devices (leaf). Thus, the size of an EKB ~~, and where there are many numbers of devices to be a distributing destination of keys, the data quantity of EKB~~ further increases.

[0198] An arrangement for reducing the size of an ~~The constitution in which the reduction of data quantity of the~~ enabling key block (EKB) ~~as described is enabled~~ will be described below. FIGS. 26A and 26B show an example in which the enabling key block (EKB) is simplified according to the key distribution device.

[0199] Similar to the example of ~~It is assumed that similarly to~~ FIG. 25, a key, for example, a content key is transmitted to devices a, g, j associated with respective leaf nodes. ~~constituting a leaf.~~ As shown in FIG. 26A, ~~a tree constituted merely by a key distributing device is constructed. In this case, a tree constitution of FIG. 26B is constructed as~~ a new simplified tree is constructed, ~~constitution~~ based on the tree structure ~~constitution~~ shown in FIG. 24B. No branch is present from Kroot to Kj — so ~~, but~~ only one branch will suffice, and ~~,~~ from K root to Ka and Kg, ~~a tree of FIG. 26A having~~ a 2-branch arrangement ~~constitution~~ is constructed merely by having ~~constituting~~ a branch point at K0.

[0200] ~~As shown in FIG. 26A, a simplified tree having only K0 as a node is produced.~~ The enabling key block (EKB) for the renewal key distribution is produced on the basis of this ~~ese~~ simplified tree~~s~~. The tree shown in FIG. 26A ~~(a)~~ is a re-constructed hierarchical tree ~~re-constructed by selecting a pass constituting a 2-branch type tree with a decodable terminal node or leaf as the lowest stage to~~ that omits unnecessary nodes. ~~The enabling~~

~~key block (EKB) for distributing a renewal key is constituted on the basis of only the key corresponding to a node or a leaf of the re-constructed hierarchical tree.~~

[0201] The enabling key block (EKB) ~~(EKR)~~ described previously with reference to FIG. 25 stores data having all keys from leaf a, g, j to Kroot, but the simplified EKB stores encrypted data with respect to only the nodes of ~~constituting~~ the simplified tree. As shown in FIG. 26B, the tag has a 3-bit structure~~constitution~~. A first bit and a second bit have meaning similar to that of the example of FIG. 25, in which if data are present in the directions of left (L) and right (R), it indicates 0, and if not, 1. A third bit is a bit for indicating ~~that~~ whether or not an encrypted key is contained in the EKB, and if data is stored, 1 appears, and if not, 0 appears.

[0202] Thus, ~~A~~an enabling key block (EKB) provided for a device (leaf) stored in a data communication network or a memory medium is considerably reduced in size ~~data quantity~~ as shown in FIG. 26B, as compared with the EKB ~~constitution~~ shown in FIG. 25. Each deice which receives~~d~~ the enabling key block (EKB) shown in FIGS. 26A and 26B sequentially decrypts only data in a portion where 1 is stored in the third bit of the tag ~~to enable realization of decrypting of a predetermined encrypted key~~. For example, the device a decrypts Enc(Ka, K(t)0) by a leaf key Ka to obtain a node key K(t)0, and decrypts encrypted data Enc(K(t)0, K(t)root) by a node key K(t)0 to obtain K(t)root. The device j decrypts encrypted data Enc(Kj, K(t)root) by a leaf key Kj to obtain K(t)root.

[0203] As described above, ~~T~~the enabling key block (EKB) is produced using only ~~the keys of leaf and node which constructs~~ a simplified new tree ~~constitution constituted merely by the device of the distributing destination to constitute a constructed tree~~ to thereby enable producing an enabling key block (EKR) with less size~~data quantity~~, whereby ~~and~~ the data distribution of the

enabling key block (EKB) can be executed efficiently.

~~[Key distributing constitution by simplified EKB (2)]~~

[0204] An arrangement ~~The constitution~~ will now be described in which the enabling key block (EKB) produced on the basis of the simplified tree shown in FIGS. 26A and 26B are further simplified to enable a further reduction of EKB size ~~data quantity~~ and allow for more efficient processing.

[0205] As described above, ~~The constitution described~~ with reference to FIGS. 26A and 26B a simplified tree is constructed ~~is the re-constructed hierarchical tree reconstructed by selecting a pass constituting a 2-branch type tree with the decodable terminal node or leaf as the lowermost stage to~~ by omitting unnecessary nodes. The structure of the enabling key block (EKB) for distributing a renewal key is based on this simplified tree~~constituted on the basis of only the key corresponding to a node or a leaf of the re-constructed hierarchical tree~~.

[0206] The simplified ~~re-constructed~~ hierarchical tree shown in FIG. 26A distributes the enabling key block (EKB) shown in FIG. 26B to enable devices a, g and j to obtain ~~obtaining~~ the renewal root key Kroot~~in the leaf a, g, j~~. In processing the enabling key block (EKB) of FIG. 26B , the device ~~leaf~~ j is possible to obtain the root key, ~~(~~K(T)root, by a one time decrypting process of Enc(Kj, K(t)root). However, the device ~~leaf~~ a~~,~~ and g obtain K(t)0 by first decrypting ~~of~~ Enc(Kg, K(t)0), and then ~~thereafter further executes~~ decrypting ~~process of~~ Enc(K(t)0, K(t)root) to finally obtain the ~~a~~ root key K(t)root. That is, devices ~~the leaf~~ a~~,~~ and g ~~are necessary to~~ execute the decrypting process twice.

[0207] In the simplified, ~~reconstructed re-constructed~~ hierarchical tree of FIGS. 26A and 26B, where the node K0 executes its own control as a control node of lower leaves ~~f~~ Ka and K~~,~~ g, for example, node K0 ~~where~~ executes control ~~of lower leaf~~ as a sub-root node ~~described later~~.~~,~~ ~~i~~It may be ~~is~~ effective

to confirm ~~in a sense of confirming~~ that the devices ~~leaf~~ a~~,~~ and g obtained the ~~a~~ renewal key. ~~, but,~~ However, where the node ~~encode~~ K0 does not carry out control of the lower leaf, or where even if the control is carried out, distribution of a renewal key from an ~~the~~ upper node is allowed, the simplified ~~re-constructed hierarchical~~ tree shown in FIG. 26A may be further simplified to omit the key of node K0~~ to produce the enabling key block (EKB) for distribution~~.

[0208] FIGS. 27A and 27B show the further simplified tree and a structure ~~the constitution~~ of the resulting enabling key block (EKB), respectively~~ as described above~~. It is again assumed ~~similarly to FIGS. 26A and 26B that~~ a key, for example, a content key, is transmitted to the devices a, g~~,~~ and j~~ constituting a leaf~~. As shown in FIG. 27A, a simplified tree is constructed in which a root Kroot and leaf nodes Ka, Kg~~,~~ and Kj are connected directly.

[0209] As shown in FIG. 27A, a further simplified tree with the ~~having a~~ node K0 omitted from the re-constructed hierarchical tree shown in FIG. 26A is produced. The enabling key block (EKB) for distributing a renewal key is produced on the basis of this~~ese~~ simplified tree~~s~~. The tree shown in FIG. 27A is ~~a re-constructed hierarchical tree~~ re-constructed merely ~~by a pass~~ for directly connecting a decodable leaf and a root.~~,~~ The enabling key block (EKB) for distributing a renewal key is formed ~~constituted~~ on the basis of a key corresponding to a leaf of the re-constructed hierarchical tree.

[0210] Although the example of FIG. 27A is an example of the arrangement ~~constitution~~ in which a terminal is a leaf, it is possible, in the ~~a~~ case of distributing keys to the uppermost node or a plurality of middle and lower nodes, to produce the enabling key block (EKB) on the basis of the simplified tree in which the uppermost node and the middle and lower nodes are directly connected to execute key distribution. As described

above, the simplified ~~re-constructed hierarchical~~ tree has a structure ~~constitution~~ in which a top node is directly connected to ~~constituting the simplified tree,~~ a terminal node or leaf node. ~~constituting the simplified tree are directly connected.~~ In the simplified tree, it is possible to structure ~~constitute~~ it as a tree having not only two branches from the top node, but a multi-branch arrangement of not less than three branches according to the number of distribution nodes or leaves.

[0211] As described above, the ~~The~~ enabling key block (EKB) of ~~described previously with reference to~~ FIG. 25 comprises encrypted ~~has the constitution in which~~ data for ~~having~~ all keys from each leaf Ka, Kg~~,~~ and Kj to Kroot. ~~encrypted are stored, and the enabling key block (EKB) stores K0 as a common node of leaf keys a, g of leaf a, g, j, and a root key, but~~ In contrast, the enabling key block (EKB) based on the simplified hierarchical tree shown in FIG. 27A omits a key of node K0, and therefore, the size of the enabling key block (EKB) of FIG. 27B is smaller than that shown in FIG. 25B~~with less data quantity is obtained, as shown in FIG. 27B~~.

[0212] The enabling key block (EKB) shown in FIG. 27B has a tag of 3 bits similar~~ly~~ to the enabling key block (EKB) shown in FIG. 26B. In the ~~a~~ first and the ~~a~~ second bits, if data are present in the directions of left (L) and right (R), it indicates 0, and if not, a 1. A third bit is ~~a bit~~ for indicating whether or not an encrypted key is stored within the EKB, and where data is stored, a 1 appears, and if not, a 0 appears.

[0213] In the enabling key block (EKB) of FIG. 27B, each device ~~leaf~~ a, g~~,~~ and j may ~~is possible to~~ obtain a root key K(t)root by a one-time decrypting process of Enc(Ka, K(t)root), or Enc(Kg, K(t)root) Enc(Kj, K(t)root).

[0214] As described above, the ~~The~~ enabling key block (EKB) produced on the basis of a simplified ~~the~~ tree ~~having the constitution~~ in which the uppermost node is directly connected to

a ~~of the simplified re-constructed hierarchical tree, the~~ terminal node ~~constituting a tree~~ or a leaf node are formed ~~directly connected is constituted~~ on the basis of only the key corresponding to the top node and the terminal node or the leaf node of the simplified ~~re-constructed hierarchical~~ tree.

[0215] As described above, the size of an EKB can be reduced by using a simplified tree as shown in either ~~A simplified new tree constitution constituted merely by a device of distributing destination, and the enabling key block (EKB) is produced using only the leaf constituting the constructed tree or only the key of node common to a leaf, as in the enabling key block (EKB) described with reference to~~ FIGS. 26A and 26B or FIGS. 27A and 27B~~, to thereby make it possible to produce the enabling key block (EKB) with less data quantity and to effectively execute data distribution of the enabling key block (EKB)~~.

[0216] The simplified hierarchical tree structure ~~constitution~~ can be utilized effectively, particularly in the EKB control arrangement ~~constitution~~ in an entity unit described below~~later~~. An ~~The~~ entity is a gathering block of a plurality of nodes or leaves of a tree~~leaf selected from a node or a leaf constituting a tree constitution as a key distribution constitution~~. The entity is set as the gathering set according to the kind of devices, or set as the gathering of a variety of forms such as a processing unit, a control unit, or a service providing unit having a common point such as control units of a device providing maker, a content provider, a settlement organization or the like. Devices classified into categories are gathered in a single entity. For example, a simplified tree similar to that described above is re-constructed by a top node (sub-roots) of a plurality of entities to produce an EKB ~~to~~ thereby. This makes it possible to produce and distribute the decodable simplified enabling key block (EKB) ~~in the device~~ belonging to the selected entity. The control structure ~~constitution~~ of the entity unit will be

described in detail later.

[0217] Such an enabling key block (EKB) as described above can ~~be constituted to~~ be stored in an information recording medium such as an optical disk, DVD or the like. For example, ~~there can be provided the constitution in which~~ an information recording medium stores an EKB and encrypted ~~, in which~~ message data ~~such as contents~~ encrypted by a renewal node key that is stored in an ~~the~~ enabling key bock (EKB). The EKB comprises ~~containing data part constituted by~~ the aforementioned encrypted key data and a tag part as position discrimination data for ~~date in~~ the associated hierarchical tree structure ~~of encrypted key data, is provided for each device~~. A destination ~~The~~ device sequentially extracts and decrypts the encrypted key data contained in the stored enabling key block (EKB) in accordance with the discrimination data of the tag part. Of course, there can be employed an arrangement ~~the constitution~~ in which the enabling key block (EKB) is distributed through a network such as an internet.

~~[EKB control constitution of entity unit]~~

[0218] Next, a description will be made of an arrangement ~~the constitution~~ in which a node or a leaf of ~~constituting~~ a tree ~~constitution as a key distribution constitution~~ is controlled by a block as a ~~the~~ gathering of a plurality of nodes or leaves. The block as the gathering of a plurality of nodes or leaves will be hereinafter called an "entity." The entity is set as the gathering set according to the kind of devices or as the gathering of various forms such as a processing unit, a jurisdiction unit or a service providing unit having a common point such as device providing makers, a content provider or a settlement organization.

[0219] The entity will be described with reference to FIGS. 28A to 28C. FIG. 28A is a view for explaining the control arrangement ~~constitution~~ of an ~~in~~ entity unit of a tree. One entity is shown

as a triangle in the figure.~~,~~ ~~f~~For example, a plurality of nodes are contained in one~~1~~ entity 2701. FIG. 28B shows the node structure ~~constitution~~ within the ~~1~~ entity 2701. The ~~1~~ entity 2701 comprises ~~is constituted by~~ a plurality of 2-branch type trees with ~~as~~ one node as a top. The top node 2702 of ~~the~~ entity 2701 will be hereinafter called a sub-root.

[0220] The terminal of the tree are represented ~~is constituted~~ by leaves ~~a leaf~~ as shown in FIG. 28C. Each terminal ~~, that~~ is~~,~~ a device. The device belongs to any entity of ~~constituted by~~ a tree ~~with a plurality of device as a leaf and~~ having a top node ~~2702~~ which is a sub-root.

[0221] As can be observed ~~will be understood~~ from FIG. 28A, an ~~the~~ entity has a hierarchical structure. This hierarchical structure will be described with reference to FIGS. 29A to 29C.

[0222] FIG. 29A is a view for explaining the hierarchical structure in a simplified form. Entities A01 to Ann are ~~constituted in the stage~~ several stages below Kroot, entities B01 to Bnk are set below the entities A1 to An, and entities C1 to Cnq are set thereunder. Each entity has a tree shape comprising ~~constituted by plural stages of~~ nodes and leaves, as shown in FIGS. 29B and 29C.

[0223] For example, the arrangement ~~constitution~~ of the entity Bnk has a plurality of nodes to a terminal node 2812, and ~~with~~ a sub-root 2811 as a top node. This entity has a discriminator Bnk, and the entity Bnk independently executes node key control corresponding to a ~~the~~ node within the entity Bnk to thereby execute control of a lower (child) entity set with the terminal node 2812 as the ~~a~~ top node. On the other hand, the entity Bnk is under the (host) entity ~~Ann~~Ann wherein the sub-root 2811 is ~~having the sub-node as~~ a terminal node of entity Ann~~2811~~.

[0224] The arrangement ~~constitution~~ of ~~an~~ entity Cn3 has a plurality of nodes and leaves as shown in FIG. 29C ~~terminal node~~ of which node 2852 is a terminal node and ~~2852 which is each~~

~~device with a~~ sub-root 2851 ~~a~~is a top node~~, and a plurality of nodes and leaves to a leaf in this case, as shown in FIG. 29C~~. This entity has a discriminator Cn3, the entity Cn3 independently executes control of a node key and a leaf key ~~corresponding to the node and leaf~~ within the entity Cn3 to thereby execute control of a leaf (device) corresponding to the terminal node 2852. On the other hand, the entity Cn3 is under the (host) entity Bn2, wherein ~~having~~ the sub-root 2851 ~~a~~is a terminal node thereof. The key control in each entity is, for example, a key renewing process, a revoke process and the like, which will be described in detail later.

[0225] ~~In a~~A device, which is a leaf of the lowest entity, ~~are~~ stores~~d~~ a node key of each node and a corresponding leaf key positioned in a pass from the ~~a~~ leaf key of the device ~~entity to which the device belongs~~ to a sub-root node, which is a top node of the entity to which the device ~~itself~~ belongs. For example, the device of the terminal node 2852 stores keys from the terminal node (leaf) 2852 to the sub-root node 2851.

[0226] An ~~The constitution of the~~ entity will be further described with reference to FIGS. 30A and 30B. The entity is able to have a tree structure having ~~constituted~~ by a variety of stage numbers. The stage number, that is, the depth, can be set according to the number of child entities corresponding to the terminal node (or leaf node (device)) controlled by the entity~~, or the device number as a leaf~~.

[0227] An arrangement ~~The detail of the constitution~~ of host and child entities ~~a~~is shown in FIG. 30A and ~~is as shown in~~ FIG. 30B, The root entity is an entity in the uppermost stage having a root key. Entities A, B, C are set as a plurality of child entities in the terminal node of the root entity, and an entity D is set as a child entity of entity C. An entity (e.g., C2901) has not less than one ~~node of the~~ terminal node as a sub-node (e.g., node 2950). Entity control may be ~~, and where entities controlled by~~

~~itself are~~ increased. For example, an entity C'2902 having plural stages of trees is newly installed with a reserve node 2950 as a top node to thereby provide ~~increase~~ control of terminal nodes 2970. As can be observed, and a child entity ~~increased~~ can be added to a ~~the control~~ terminal node.

[0228] A ~~The~~ reserve node will be further described with reference to FIG. 31. Entity A, 3011, controls ~~has~~ child entities B, C, D ...~~ to be controlled~~, and has one reserve node 3021. Where it is desired to increase the number of child entities that are controlled, ~~child entities to be controlled are further increased,~~ a child entity e.g., A', 3012, ~~under the own control~~ is set to the reserve node, e.g., 3021. Similarly,~~ and~~ child entities F~~,~~ and G to be controlled can be further set to the terminal node of the child entity A', 3012. Also in the child entity A', 3012~~ under the own control~~, at least one of the terminal nodes is set as a reserve node 3022 whereby another child entity e.g., A"3013 can be ~~is~~ further set~~ to further increase the control entities~~. One, or more, reserve nodes are secured also in the terminal node of the child entity A"3013. This use of ~~Such a~~ reserve nodes allows ~~holding constitution as described is employed whereby the~~ child entities to ~~under a certain entity can~~ be increased endlessly. With respect to the reserve node~~entity~~, not only one terminal node but a plurality of nodes may be set as a reserve node.

[0229] In the respective entities, the enabling key block (EKB) is formed ~~constituted~~ in the entity unit, and key renewing and revoke processing are to be executed in the entity unit. As shown in FIG. 31, the enabling key block (EKB) of an individual entity is set to a plurality of entities A, A', A", but these can be collectively controlled, for example, by device makers who controls the entities A, A', A" in common.

~~[Registration process of new entities]~~

[0230] Next, the registration process of new entities will be described. FIG. 32 shows a registration processing sequence. ~~A description will be made in accordance with the sequence in FIG. 32.~~ A newly added (child) entity(N-En) provides a request for a ~~newly added during the constitution of a tree executes requesting of~~ new registration to a host entity (P-En). Each entity holds a public key in accordance with a public key encryption system, and a new entity sends its own public key to the host entity (P-En) when a registration request is made.

[0231] The host entity (P-En), which receives~~d~~ the registration request, transfers the received ~~a~~ public key of the new ~~a~~ (child) entity ~~received~~ to a certificate authority (CA) and receives back a public key certificate for ~~of~~ the new (child) entity (N-En) to which a signature of CA is added. These procedures are carried out as a procedure for mutual authentication between the host entity (P-En) and the new (child) entity (N-En).

[0232] When the authentication procedure ~~of the new registration requesting entity~~ is successfully terminated, the host entity (P-En) ~~grants the registration of the new (child) entity (N-En) to~~ transmits a node key (of the new (child) entity (N-En)) to the new (child) entity (N-En)~~.~~. This node key is a ~~one~~ node key of the terminal node of the host entity (P-En) which corresponds to a top node of the new (child) entity (N-En), that is, a sub-root key.

[0233] When the transmission of the node key is finished, the new (child) entity (N-En) constructs the tree structure ~~constitution~~ of the new (child) entity (N-En), sets a sub-root key of a top node received to a top of the constructed tree, and sets node and leaf keys to produce an enabling key block (EKB) within the entity. The enabling key block (EKB) within one entity is called a sub-EKB.

[0234] On the other hand, the host entity (P-En) produces the sub-EKB within the host entity(P-En) to which is added a terminal

node to be enabled by the addition of the new (child) entity (N-En).

[0235] When the sub-EKB comprises ~~constituted by~~ a node key and a leaf key within the new (child) entity (N-En) is produced, the new (child) entity (N-En) transmits it to the host entity (P-En).

[0236] The host entity (P-En) which receives the sub-EKB from the new (child) entity (N-En) transmits the received sub-EKB and a renewal sub-EKB of the host entity (P-En) to a key distribute center (KDC) .

[0237] The key distribute center (KDC) is able to produce various EKBs, that is, an EKB that can be decrypted merely by a specific entity or device on the basis of sub-EKBs of all entities. An EKB to which such a decodable entity or device is set is distributed, for example, to a content provider, who encrypts a content key on the basis of the EKB to distribute it through a network or store it in a recording medium, thus enabling distribution of a content for use ~~that can be used merely~~ by a specific device.

[0238] The registration processing with respect to the key distribute center (KDC) of the sub-EKB of the new entity is not limited to a method for sequentially transferring the sub-EKB through the host entity. For example, ~~but there can be also employed the constitution which executes~~ the processing for registering the sub-EKB in the key distribute center (KDC) can be performed directly from the new registration entity without the intervention of the host entity. The correspondence of the host entity to a newly added ~~the~~ child entity ~~to be newly added to the host entity~~ will be described with reference to FIG. 33. One terminal node 3201 of the host entity serves ~~is distributed~~ as a top node of the newly added child entity, ~~to the child entity~~ whereby the child entity is added as an entity under the control of the host entity. This control includes the ability to perform remote processing with respect to the child. ~~The entity under the~~

~~control of the host entity termed herein, which will be described later, also includes meaning of the constitution in which the revoke processing of the child entity can be executed by the host entity.~~

[0239] As shown in FIG. 33, when a new entity is set to the host entity, one node ~~3201~~ of a terminal node (e.g., node 3201), which is a leaf node of the host entity and a top node (e.g., node 3202) of the newly added entity are set as equal nodes. That is, a ~~one~~ terminal node, which is a ~~one~~ leaf node of the host node, is set as a sub-root of the newly added entity. By being so set, the newly added entity is enabled under the whole tree structure~~constitution~~.

[0240] FIGS. 34A and 34B show ~~an~~ examples of a renewal EKB that is produced by the host entity when the newly added entity is set. FIG. 34A shows an example of a sub-EKB produced by the host entity when a new entity is added to terminal node (node 100) 3303 of the host entity~~is applied to the newly added entity~~. ~~, i~~In the arrangement ~~constitution~~ shown in FIG. 34A, the host entity ~~which~~ has a terminal node (node 000) 3301 ~~which has been effectively present~~ and a terminal node (node 001) ~~node~~ 3302.

[0241] The sub-EKB has the form ~~constitution~~ as shown in FIG. 34B. The sub-EKB comprises ~~There are~~ a host node key (encrypted by a terminal node which has been effectively present), a further host node key (encrypted by a ~~the~~ host node key), ... and a sub-root key. Similar~~ly~~ to FIG. 34B, each entity has and controls an EKB that is structured ~~constituted~~ to have a host node encrypted by an effective terminal node or leaf key, ~~encrypts~~ a further host node key encrypted by a ~~the~~ host node key, and ~~an encrypted data to~~ a sub-root key~~sequentially being increased in depth~~.

~~[Revoke processing under the control of entity]~~

[0242] Next, a description will be made of the revoke processing of a device or an entity in an arrangement ~~the constitution~~ in which the key distribution tree structure ~~constitution~~ is

controlled as an entity unit. As described earlier with respect to FIGS. 3 and 4, it is possible to revoke a device and distribute an EKB that is only decodable by the valid destination device. ~~In previous FIGS. 3 and 4, a description has been made of the processing for distributing an enabling key block (EKB) in which only the specific device out of the whole tree constitution is decodable, and the revoked device is undecodable.~~ The revoke processing described with respect to ~~in~~ FIGS. 3 and 4 is the processing for revoking a specific device ~~which is a specific leaf~~ out of the whole tree. However, ~~but the constitution by~~ entity control makes it ~~of the tree is~~ possible to execute ~~the~~ revoke processing for every entity.

[0243] A description will be made hereinafter of ~~the~~ revoke processing with respect to ~~in the constitution under the~~ entity control with reference to FIGS. 35A to 35D and drawings continuous thereto. ~~FIGS. 35A to 35D is a view for explaining the revoke processing of a device by an entity which controls an entity in the lowest stage, out of entities constituting a tree, that is, an entity controlling individual devices.~~

[0244] FIG. 35A shows the key distribution tree structure comprising entities~~under the control of entity~~. A root node is set to the uppermost part of the tree to which are coupled ~~, and~~ entities A01 to Ann.~~,~~ ~~e~~Entities B01 to Bnk are below the ~~previous~~ entities A01 to Ann, and ~~entities C1 to cn in~~ the lowest stage comprises entities C1 to Cn ~~are constituted~~. In the lowest entity, the terminal nodes (leaves~~f~~) are ~~is~~ individual devices, for example, a recording and reproducing unit, a reproducing exclusive-use unit or the like.

The revoke processing is independent~~ly~~ in each entity. For example, in the entities C1 to Cn~~ in the lowest stage~~, the revoke processing of a device ~~of a leaf~~ is executed. FIG. 35B shows the tree structure ~~constitution~~ of an entity Cn, 3430, which is one of the entities in the lowest stage. The entity Cn, 3430, has a

top node 3431, and leaves (terminal nodes) associated with ~~a leaf which is a terminal node has~~ a plurality of devices.

[0245] Assume that a device is to be revoked, for example, a device 3432 ~~is present in a leaf,~~of the entity Cn~~,~~ 3430. The latter produces an enabling key block (sub-EKB) ~~constituted by~~having a node key and a leaf key in the independently renewed entity Cn. This enabling key block is a key block ~~constituted by~~comprising an encrypted key that cannot be decrypted ~~in the revoke key~~ in the ~~revoke~~revoked device 3432~~ but that can be decrypted by only the device constituting other leaf~~. A controller of the entity Cn produces ~~it as a~~this renewed sub-EKB. ~~Concretely, the block, which~~The renewed sub-EKB comprises an encrypted key which renews node keys of nodes 3431, 3434, and 3435 ~~constituting a pass associated with a~~on the path from the sub-root to ~~a revoke~~revoked device 3432.~~, and can decrypt the renewal key~~ As such only ~~in~~ a leaf device other than the ~~revoke~~revoked device 3432 can decrypt the renewal sub-EKB. This processing corresponds to the processing ~~in which a root key is replaced by a sub-root which is a top key of entity, in the revoke processing constitution~~ described in association with FIGS. 3 and 4.

[0246] The enabling key block (sub-EKB) renewed by the entity Cn, 3430 ~~through the revoke processing~~ is transmitted to the host entity. In this case, the host entity is an entity Bnk, 3420, ~~and an entity having a top node 3431 of~~ in which terminal node 3431 serves as the top node of the entity Cn, 3430~~ as a terminal node~~.

[0247] The entity Bnk, 3420, ~~when~~ receives the enabling key block(sub-EKB) from the child entity Cn, 3430, sets the terminal node 3431 of the entity Bnk, 3420, (corresponding to the top node 3431 of the entity Cnk, 3430 contained in the key block) to a key renewed in the child entity Cn, 3430, and executes the renewal processing of sub-EKB ~~of own entity Bnk, 3420~~for itself. FIG. 35C shows the tree of entity Bnk, 3420. In the entity Bnk, 3420, a

node key to be renewed is a node key on a ~~pass~~path from the sub-root 3421 in FIG. 35C to the terminal node 3431, ~~constituting an~~which is associated with the entity containing ~~a revoke~~the revoked device. ~~That is~~In this example, node keys of the nodes 3421, 3424, and 3425~~ constituting a pass associated with the node 3431 of the entity transmitted from the renewal sub-EKB are to be renewed~~. These node keys ~~of nodes~~ are renewed to produce a new renewal sub-EKB of the entity Bnk, 3420.

[0248] Further, the enabling key block (sub-EKB) renewed by the entity Bnk, 3420 is transmitted to the host entity. In this case, the host entity is the entity Ann, 3410, ~~and an entity having a top node 3421 of~~ in which terminal node 3421 serves as the top node of the entity Bnk, 3420~~ as a terminal node~~.

[0249] The entity Ann, 3410, ~~when~~ receives the enabling key block (sub-EKB) from the child entity Bnk, 3420, sets the terminal node 3421 of the entity Ann, 3410 (corresponding to the top node 3421 of the entity Bnk, 3420 contained in the key block) to a key renewed in the child entity Bnk, 3420, and executes the renewal processing of sub-EKB~~ of own entity Ann, 3410~~ for itself. FIG. 35D shows the tree of entity Ann, 3410. In the entity Ann, 3410, node keys to be renewed are node keys 3411, 3414, 3415 on a ~~pass~~path from the sub-root 3411 in FIG. 35D to the terminal node 3421, ~~constituting an~~ which is associated with the entity containing ~~a revoke~~ the revoked device. These node keys ~~of nodes~~ are renewed to produce a new renewal sub-EKB of the entity Ann, 3410.

[0250] These processes sequentially execute in the host entity to the root entity described in association with FIG. 30B. The revoke processing of devices is completed by a series of processes as described. The sub-EKB renewed in the entity is finally transmitted to the key distribute center (KDC) and stored therein. The key distribute center (KDC) produces various EKBs on the basis of the renewal sub-EKB of all entities. The renewal EKB

is an encrypted key block that cannot be decrypted by the revoked device~~ revoked~~.

[0251] FIG. 36 shows a revoked process sequence~~ of revoke process of the device. The processing procedure will be described with reference to the sequence figure of FIG. 36. ~~First, the device control entity (D-En) in the lowest stage of the tree ~~constitution ~~carries out a key ~~renewing~~renewal necessary for revoking a leaf ~~to be revoked ~~in the device control entity (D-En) to produce a new sub-EKB of the device control entity (D-En)~~.~~. The sub-EKB is sent to the host entity. The host entity (P1- En), which received the renewal sub-EKB (D), produces a renewal sub-EKB (P1) in which a terminal node key (corresponding to a renewal top node of the ~~renewal ~~renewed sub-EKB (D)) is renewed ~~and~~along with node keys on a pass from the terminal node to the sub-root. These processes are sequentially executed in the host entity, and all sub-EKBs finally renewed are stored and controlled by the key distribute center (KDC).

[0252] FIGS. 37A and 37B show an example of an enabling key block (EKB) to be produced as a result~~ that the host entity carries out renewal processing by the revoke processing of a device~~of revoking a device.

[0253] FIGS. 37A and 37B are views ~~each ~~for explaining an example of an EKB produced in the host entity, which received a renewal sub-EKB from ~~the ~~a child entity containing a ~~revoke~~ revoked device~~,~~. ~~In the constitution shown in~~ In FIG. 37A~~.~~, ~~A~~a top node of the child entity containing the ~~revoke~~revoked device corresponds to a terminal node (node 100) 3601 of the host entity.

[0254] The host entity renews those node keys ~~which~~that are present in a pass (path) from the sub-root of the host entity to the terminal node (node 100) 3601 to produce a new renewed sub-EKB. The ~~renewal~~renewed sub-EKB is as shown in FIG. 37B. ~~The~~A renewed key is shown in FIG. 37B with ~~the~~an underline and [']

attached thereto. ~~The node keys on a pass from the renewed terminal node to the sub-rot are renewed to obtain a renewal sub-EKB in its entity.~~

[0255] Next, ~~processing where an object subjected to revoking is an entity, that is,~~ revoke processing of entity~~,~~ will be described.

[0256] FIG. 38A shows ~~the~~a key distribution tree structure ~~by~~under entity control. A root node is set to the uppermost part of the tree, and entities A01 to Ann ~~are constituted~~have several stages thereunder~~,~~. In particular, entities B01 to Bnk ~~are constituted in~~represent the stage ~~lower than the former~~below entities A01 to Ann, and entities C1 to cn ~~are constituted in~~ represent the stage ~~lower than the further stage are constituted~~below entities B01 to Bnk. In the lowest entity, the terminal node (leaf) is an individual ~~devices~~device, for example, such as recording and reproducing unit, a reproducing exclusive-use unit or the like.

[0257] Now, a description is made of the ~~case where~~ situation in which the revoke processing is carried out with respect to the entity Cn, 3730. The entity Cn, 3730 ~~in the lowest stage~~ has ~~the constitution in which~~ a top node ~~3431~~3731 ~~is provided~~, and a plurality of devices are provided on ~~a leaf which is a terminal node~~leaves (terminal nodes), as shown in FIG. 38B.

[0258] The revoking of the entity Cn, 3730, ~~enables collective~~provides the ability to revoke ~~of~~ all devices belonging to the entity Cn, 3730 from the tree structure. The revoke processing of the entity ~~cn~~Cn, 3730 is executed in the entity Bnk, 3720, which is the host entity of the entity Cn, 3730. The entity Bnk, 3720, is an entity ~~having the top node 3731~~in which a terminal node 3731 is a top node of the entity Cn, 3730 ~~as a terminal node~~.

[0259] Where revoking of the child entity Cn, 3730 is executed, the entity Bnk, 3720 renews a terminal node 3731 of the entity

Bnk, 3720, corresponding to the top node 3731 of the entity Cnk, 3730, and further carries out renewing of node keys on a ~~pass~~path from the ~~revoke~~revoked entity 3730 to the sub-root of the entity Bnk, 3720, to ~~produce an enabling key block to~~ produce a ~~renewal~~ renewed sub-EKB. ~~The node key to be renewed is a node key on a pass from the sub-root 3721 shown in FIG. 38C to a top node of a revoke entity.~~ That is, nodes 3721, 3724, 3725 and 3731 are objects to be renewed. These node keys ~~of nodes~~ are renewed to produce a new ~~renewal~~renewed sub-EKB of the entity Bnk, 3720.

[0260] Alternatively, ~~where revoking of the~~in performing revocation in a child entity, Cn, 3730~~ is executed~~, the entity Bnk, 3720 does not renew the terminal node 3731 ~~of the entity Bnk, 3720~~ corresponding to the top node ~~3731~~ of the entity Cnk, 3730, ~~but~~and only renews ~~a node key except the terminal node 3731 on the pass from the revoke entity 3730 to the sub-root of the entity Bnk, 3720 to produce an enabling key block~~ nodes 3721, 3724, and 3731 to produce a renewal sub-EKB.

[0261] Further, the enabling key block (sub-EKB) renewed by the entity Bnk, 3720 is transmitted to the host entity. In this case, the host entity is an entity Ann, 3710, which is an entity having a top node 3721 of the entity Bnk, 3720 as a terminal node.

[0262] When an enabling key bock (sub-EKB) is received from the child entity Bnk, 3720, the entity Ann, 3710, sets the terminal node, 3721, of the entity Ann, 3710, (corresponding to the top node 3721 of the entity Bnk, 3720) ~~contained in the key block~~ to a key renewed in the child entity Bnk, 3720 ~~to execute~~and executes renewal processing of the sub-EKB ~~of the own entity Ann, 3710~~for itself. FIG. 38D shows the tree ~~constitution~~structure of the entity Ann, 3710. In the entity Ann, 3710, the node key to be renewed is a node key of each node 3711, 3714, and 3715 constituting a ~~pass~~path from the sub-root 3711 ~~of FIG. 38D~~ to the node 3721 of the entity having transmitted the renewal sub-EKB. These node keys ~~of the nodes~~ are renewed to produce a new renewal

sub-EKB of the entity Ann, 3710.

[0263] These processes are sequentially executed in the host entity ~~to execute it to the root entity~~ described with reference to FIG. 30B, above. The revoke processing is completed by a series of processes. The sub-EKB renewed in the respective entity is finally transmitted to the key distribute center (KDC) and stored. The key distribute center (KDC) produces various EKBs on the basis of the renewal sub-EKB of all entities. The renewal EKB is an encrypted key block that cannot be decrypted by the device belonging to the entity revoked.

[0264] FIG. 39 shows a revoke processing sequence ~~of revoke process of the~~ for an entity. ~~The processing procedure will be described with reference to the sequence figure of FIG. 39.~~ First, the entity control entity (E-En) produces a renewed sub-EKB which revokes ~~for revoking the entity carries out key renewing necessary for revoking~~ a terminal node ~~to be revoked in the entity control entity (E-En) to produce a new sub-EKB of the entity control entity (E-En)~~. The renewed sub-EKB is sent to the host entity. The host entity (P1- En), which received the renewed ~~al~~ sub-EKB, ~~(E)~~ produces a renewed~~al~~ sub-EKB (P1) in which a terminal node key (corresponding to a renewal top node of the entity (E-En)) ~~renewal sub-EKB (P1)~~ is renewed and node keys on a path ~~pass~~ from the terminal node to the sub-root are also renewed. These processes are sequentially executed in the host entity, and all sub-EKBs finally renewed are stored and controlled by the key distribute center (KDC). The key distribute center (KDC) produces various EKB on the basis of the renewal EKB of all entities. The renewal EKB is an encrypted key block that cannot be decrypted by a device belonging to a revoked ~~the~~ entity ~~revoked~~.

[0265] FIG. 40 is a view illustrating ~~for explaining~~ the correspondence of a revoked ~~the~~ child entity ~~revoked~~ to the host entity which carried out the revoking process. ~~A terminal node~~

~~3901 of the host entity is renewed by revoking the entity, and a new sub-EKB is produced by renewing of~~ In performing the revoking process, the host entity renews terminal node 3901 and also renews those node keys that ~~which~~ are present in a path ~~pass~~ from the terminal node 3901 to the sub-root in the tree of the host entity to produce a new sub-EKB. As a result, the node key of the top node 3902 of the revoked child entity ~~revoked~~ does ~~is~~ not coincide~~d~~ with the node key of the terminal node 3901 of the host entity. ~~EKB produced by the key distribute center (KDC) a~~After revoking of the entity, an EKB produced by the key distribute center (KDC) is ~~to be~~ produced on the basis ~~of the key~~ of the renewed terminal node ~~renewed,~~. ~~and t~~Therefore, the device corresponding to the leaf of the child entity not holding the renewal key is disabled~~s~~ from decrypting those subsequent ~~of~~ EKBs produced by the key distribute censer (KDC).

[0266] While in the foregoing, the revoking process has been described in the context of revoking the ~~of the~~ entity in the lowest stage ~~for controlling the device has been described~~, processing for ~~revoking the entity control~~ an entity in the middle stage of the tree ~~by the host entity~~ is also enabled by a similar ~~the~~ process ~~similar to that described above~~. By revoking an ~~the~~ entity ~~control entity~~ in the middle stage, a plurality of entities and devices belonging to ~~the~~ lower levels of the tree ~~entity control entity revoked~~ can be collectively revoked ~~collectively~~.

[0267] As described above, the process for revoking an entity is similar to that for revoking a single device. ~~by the execution of revoking in an entity unit, revoking process which is simple as compared with the revoking process for executing it in a device unit one by one becomes enabled.~~

~~[Capability control of entity]~~

[0268] Next, a description will be made of a ~~the~~ processing arrangement ~~constitution~~ in which ~~in the key distribution tree~~

~~constitution in an entity unit, capability granted by each entity is controlled to carry out~~ content distribution is carried out by an entity in accordance with a ~~according to the~~ capability. The term "capability" refers to~~capability termed herein is~~, for example, a defined ~~information of the~~ data processing ability of a device. For example, whether decrypting of specific compressed voice data is enabled, whether a specific voice reproducing system is enabled~~granted~~, whether ~~or~~ specific image processing program can be performed~~processed~~, or whether a device is ~~a device~~ capable of processing a ~~what~~ content or a program.

[0269] FIG. 41 shows an example of an ~~the~~ entity arrangement ~~constitution~~ which has defined capabilities~~defines the capability~~. This is a tree ~~the constitution~~ in which a root node is positioned at the uppermost top of the key distribution tree, a plurality of entities are connected to the lower layer, and each node has a 2-branch. Here, for example, an entity 4001 is defined as an entity having the capability to enable ~~grant~~ either voice reproducing systems A, B or C. ~~Concretely, for example, where music data compressed by voice compressed program A, B or C system are distributed, processing for extending the device belonging to the entity constituted below the entity 4001 is enabled.~~

[0270] Similarly, entity 4002, entity 4003, entity 4004, and entity 4005 are respectively defined as entities having the capability ~~capable~~ of using ~~processing~~ voice reproducing system B or C, voice reproducing system A or B, voice reproducing system B, and voice reproducing system C, respectively.

[0271] On the other hand, an entity 4021 is defined as an entity having the capability to enable ~~grant~~ image reproducing systems p, q~~,~~ and r.~~,~~ ~~and a~~An entity 4022 and an entity 4023 are respectively defined as entities having the capability to use ~~enable~~ image reproducing ~~of a~~ system p.

[0272] The capability information of the entities as described

is controlled in the key distribute center (KDC). For example, where a content provider desires to distribute music data compressed by a specific compression program to various devices, an enabling key block (EKB) (decodable with respect to only the device which can reproduce the specific compression program) can be produced on the basis of the capability information of each entity. The content provider ~~for distributing contents~~ distributes a content key encrypted by the enabling key block (EKB), which is produced on the basis of the capability information, and also distributes compressed voice data encrypted by the content key to the devices. As such~~By the provision of this constitution~~, it is possible to accurately provide ~~accurately a~~ data only ~~specific processing program~~ to ~~only the~~ a device capable of processing that data.

[0273] While in FIG. 41, ~~the constitution in which capability information is defined in connection with all the entities is shown,~~ it is noted that it is not ~~always~~ necessary to define the capability information with respect to all the entities ~~as in the constitution of FIG. 41~~, but ~~the constitution may be employed in which for example~~, as shown in FIG. 42, capability may be ~~is~~ defined with respect to only the entity in the lowest stage to which the device belongs.~~,~~ The capability of the device belonging to the entity in the lowest stage is controlled in the key distribute center (KDC), and the enabling key block (EKB) ~~that can be decrypted merely for the device capable of providing a process desired by a content provider~~ is produced on the basis of capability information defined in the entity in the lowest stage. FIG. 42 shows an arrangement ~~the constitution~~ in which the capability in entity 4101 ~~– 4105 for which the device is defined~~, is defined at ~~in~~ the terminal node for which the device is associated.~~,~~ The ~~and~~ capabilitiesy with respect to these entities is controlled in the key distribute center (KDC). For example, to the entity 4101 belong devices capable of processing a system B

with respect to voice reproducing and a system r with respect to image reproducing, respectively. To the entity 4102 belong devices capable of processing a system A with respect to voice reproducing and a system q with respect to image reproducing, respectively.

[0274] FIGS. 43A and 43B show an example ~~of the constitution~~ of a capability control table controlled in the key distribute center (KDC). Each row of the capability control table comprises a capability test, an entity ID, an EKB, and sub-root information. ~~The capability control table has the data constitution as shown in FIG. 43A. That is, propriety with respect to various data processes is set to [1] or [0] such that there are an entity ID as a discriminator for discriminating entities and a capability list indicative of capability defined in the entities, and i~~In the capability list, ~~as shown in FIG. 43B,~~ for example, if a voice data reproducing processing system (A) ~~is~~ can be processed, [1] appears, if not, [0] appears, and if a voice data reproducing processing system (B) can be processed, [1] appears, if not, [0] appears. The method of setting capability is not limited to such a form as described, but other arrangements ~~constitutions~~ may be employed~~ if capability with respect to the control device of entities can be discriminated~~.

[0275] For each capability test, corresponding entity ID, sub-EKB (which may be ~~In the capability control table, where sub-EKB if each entity of sub-EKB is~~ stored in a separate data base), ~~discrimination information of sub-EKB is stored,~~ and sub-root information ~~node discrimination data of each entity~~ is stored.

[0276] In the key distribute center (KDC), EKBs are produced such that ~~for example,~~ only ~~the~~ devices capable of reproducing ~~a~~ specific content can decode the respective ~~produces a decodable~~ enabling key block (EKB)s~~ on the basis of the capability control table~~. The processing for producing the enabling key block on the basis of capability information will be described with reference

to FIG. 44.

[0277] First, in Step S4301, the key distribute center (KDC) selects those entities ~~an entity~~ having the designated capability from the capability control table. ~~Concretely, f~~For example, where a content provider desires to distribute reproducible data on the basis of the voice data reproducing processing system A, ~~is set to [1] is selected from the capability list of FIG. 43A.~~ an entity, ~~for example, in which item of the voice data reproducing (system A) is set to [1],~~ is selected from the capability control table ~~list~~ of FIG. 43A in which the corresponding bit on the capability list associated with voice data producing processing system A is set to [1].

[0278] Next, in Step S4302, a list of those selected entity IDs ~~constituted by the selected entities~~ is produced. Next, in Step S4303, a path ~~pass (a pass of key distribution constitution)~~ necessary for a tree comprising the ~~constituted by~~ selected entity ID is selected. In Step 4304, a check is made to determine if all paths have been selected. ~~whether or not all pass selections contained in the list of selected entity ID are completed is judged to produce a pass in Step S4303 till completion. This means the process for sequentially selecting the respective passes where a plurality of entities are selected.~~

[0279] When all path ~~pass~~ selections ~~contained in the selected entity ID~~ are completed, the procedure proceeds to Step S4305 to form ~~constitute~~ a key distribution tree structure for ~~constituted merely by~~ the selected entities.

[0280] Next, in Step S4306, renewing of node keys of the tree structure produced in Step S4305 is carried out to produce renewed node ~~renewal nod~~ keys. Further, the sub-EKB information of the selected entities ~~constituting the tree~~ is taken out of the capability control table, and an ~~the~~ enabling key block (EKB) ~~that can be decrypted merely in the device of the selected entities~~ is produced on the basis of the sub-EKB and the

renewed~~al~~ node key produced in Step S4306. The enabling key block (EKB) thus produced is utilized only ~~merely~~ in the device having the specific capability.~~, that is, being a decodable enabling key block (EKB.~~ For example, a content key is encrypted by the enabling key block (EKB), and ~~a~~ content compressed on the basis of a specific program in the content key is distributed to the device, whereby the content is utilized only in the specific ~~decodable~~ device selected by the key distribute center (KDC).

[0281] As described above, in the key distribute center (KDC), the capability control table is used to select ~~for example,~~ only those ~~the~~ devices capable of reproducing the specific content and only those selected devices can decode the ~~produces the decodable~~ enabling key block (EKB) ~~on the basis of the capability control table~~. Accordingly, where a new entity is registered, it is necessary to ~~previously~~ obtain the capability of a newly registered entity. This process ~~The processing of notifying capability with the entity new registration~~ will be described with reference to FIG. 45.

[0282] FIG. 45 shows a sequence for providing ~~is a view showing a~~ capability notice for a ~~processing sequence where the~~ new entity ~~is participated in the key distribution tree constitution~~.

[0283] The new (child) entity (N-En) added ~~newly~~ to the tree ~~constitution~~ executes a new registration request with respect to the hose entity (P-En). Each entity holds a public key in accordance with the public key encryption system, and the new entity sends its own public key to the host entity (P-En) when the registration request takes place.

[0284] The host entity (P-En) which received the registration request, transfers the received public key of the new (child) entity (N-En) ~~received~~ to the certificate authority (CA), and receives therefrom a public key of the new (child) entity (N-En) to which a signature of CA is added. These procedures are carried out as the procedure of mutual authentication between the host

entity (P-En) and the new (child) entity (N-En).

[0285] When the authentication of the new registration request entity is finished ~~by these processes~~, the host entity (P-Ne) grants the registration of the new (child) entity (N-En) and ~~to~~ transmits a node key of the new (child) entity (N-En) to the new (child) entity (N-En). This node key is one node key of the terminal node of the host entity (P-En) and corresponds to a top node of the new (child) entity (N-En), that is, a sub-root key.

[0286] When transmission of this node key is finished, the new (child) entity (N-En) constructs the tree ~~constitution~~ of the new (child) entity (N-En)~~-~~, sets the sub-root key to ~~of the top node received to~~ the top of the constructed tree, sets keys of each node and leaf, and produces the enabling key block (sub-EKB) in the entity. On the other hand, the host entity (P-En) also produces the sub-EKB in the host entity (P-En) to which is added a terminal node resulting from ~~to be effective by~~ the addition of the new (child) entity (N-En).

[0287] When the new (child) entity (N-En) produces the sub-EKB ~~constituted by a node key and a leaf key in the new (child) entity (N-En)~~, the new (child) entity (N-En) transmits it to the host entity (P-En), and further provides to the host entity ~~notifies~~ capability information ~~with~~ in connection with ~~the~~ devices controlled by ~~own~~ entity (N-En)~~to the host entity~~.

[0288] The host entity (P-En), which received the sub-EKB and the capability information from the new (child) entity (N-En), transmits the received sub-EKB, the received ~~and~~ capability information~~received~~, and the renewed sub-EKB of the host entity (P-En) to the key distribute center (KDC).

[0289] The key distribute center (KDC) registers the received sub-EKB and received capability information of the new entity ~~received~~ in the capability control table described with reference to FIGS. 43A and 43B, and renews the capability control table. The key distribute center (KDC) can ~~is possible to~~ produce various forms of EKBs, that is, an EKB that can be decrypted only ~~merely~~ by the entity having a specific capability or device~~s~~.

[0290] The present invention has been described in detail with reference to the specific embodiments. However, it is obvious that those skilled in art may amend or replace the embodiments within the scope not departing from the subject matter of the present invention. That is, the present invention has been disclosed in the form of illustration and should not be interpreted narrowly~~imitatively~~. For judging the subject matter of the present invention, reference should be made to the claims described herein after.

~~Industrial Applicability~~

[0291] As described above, according to the information processing system and method according to the present invention, in the production of an ~~the~~ enabling key block (EKB) (that can be applied as the encrypting processing key block such as a content key, an authentication key, a content check value producing key, a program data or the like), the hierarchical key distribution tree is reconstructed according to the distribution device, and the enabling key block (EKB) is produced on the basis of the node and leaf contained in a ~~the reconstructed~~ simplified tree. Therefore, a considerable reduction in the size ~~of data quantity~~ of the enabling key block (EKB) is realized.

[0292] Further, according to the information processing system and method according to the present invention, the enabling key block (EKB) is formed on the basis of a ~~the~~ simplified ~~reconstructed hierarchical~~ tree ~~is constituted~~, and data ~~for judging the propriety of encrypted key data~~ is contained in a tag

as a position discriminator of encrypted key data in the EKB. Therefore, a considerable reduction in data quantity of the EKB is realized, and extraction of encrypted key data using a tag in the device which received the EKB is facilitated to make the EKB decrypting process in the device more effective.

409691_1.DOC

Version With Markings To Show Changes Made

MARKED-UP COPY OF AMENDED ABSTRACT:

An ~~The~~ enabling key block (EKB) used in an encrypted key distributing ~~constitution of a~~ tree structure is generated by forming ~~reconstructing~~ a simplified 2-branch or multi-branch type tree with a terminal node or leaf which is capable of decrypting ~~can decrypt as the lowest stage, and~~ on the basis of a ~~only the~~ key corresponding to a node or a leaf of the simplified ~~reconstructed hierarchical~~ tree. Further, the EKB includes a tag for indicating a position of ~~a tag as discrimination data at a tree position of~~ an encrypted key in the tree~~stored in EKB is stored~~. The tag not only discriminates ~~a~~ position but also stores data for judging the presence of encrypted key data within the EKB. As such, a considerable reduction in data quantity is realized, and the decrypting process in a device is also simplified. ~~Thus, an information processing system and method capable of reducing data quantity of an enabling key block (EKB) used in an encrypted key constitution of a tree structure is realized.~~

#8

OIPE JC32
JAN 2 2 2003
PATENT & TRADEMARK OFFICE

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail, in an envelope addressed to: Commissioner for Patents, Washington, DC 20231, on the date shown below.

Dated: January 16, 2003 Signature: (Joseph J. Opalach)

Docket No.: SONYAK 3.3-161
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Ishiguro et al.

Application No.: 09/980,952

Group Art Unit: 2131

Filed: December 3, 2001

Examiner: Not Yet Assigned

For: INFORMATION PROCESSING SYSTEM AND METHOD

RECEIVED
JAN 2 4 2003
Technology Center 2100

Commissioner for Patents
Washington, DC 20231

SUBMISSION OF MARKED-UP CORRECTED DRAWINGS

Dear Sir:

Subject to the approval of the Examiner handling this application, please amend FIGS. 11B, 16, 17 and 44 in this application in the manner indicated in the attached marked-up photocopies of the original drawings, marked in red. No new matter has been added.

If there are any additional charges in connection with this requested amendment, the Examiner is authorized to charge applicant's Deposit Account No 12-1095 therefor.

Dated: January 16, 2003

Respectfully submitted,

By
Joseph J. Opalach
Registration No.: 32,768
LERNER, DAVID, LITTENBERG,
KRUMHOLZ & MENTLIK, LLP
600 South Avenue West
Westfield, New Jersey 07090
(908) 654-5000
Attorneys for Applicants

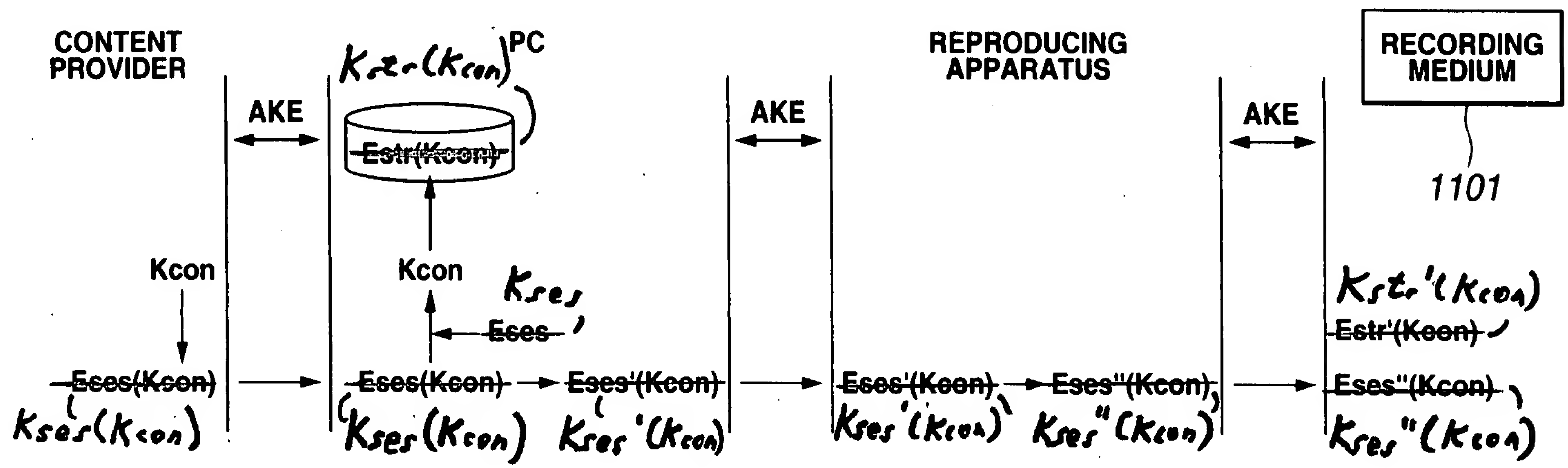

FIG.11A

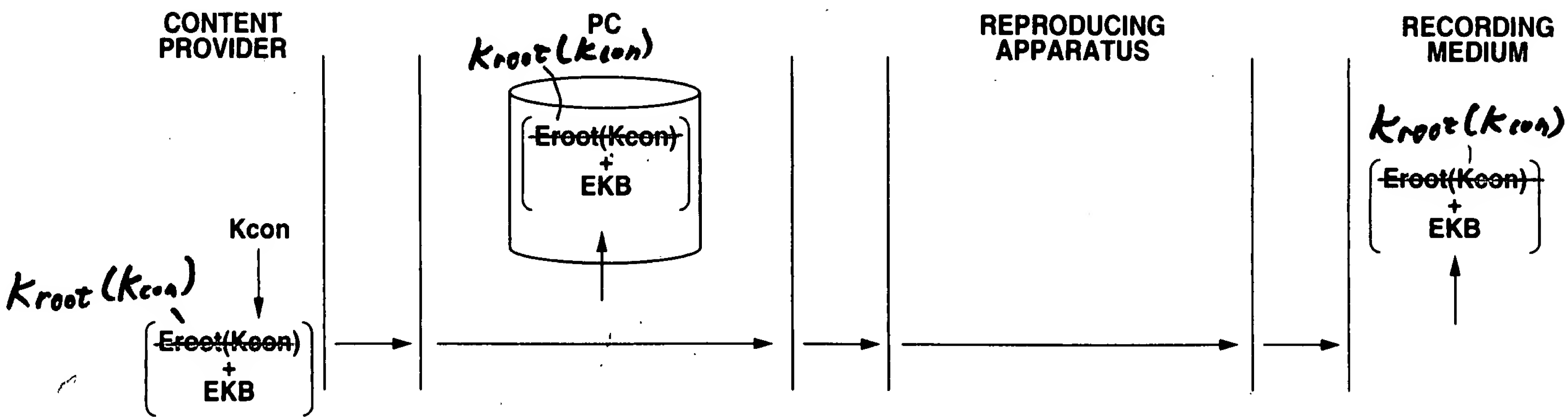

FIG.11B

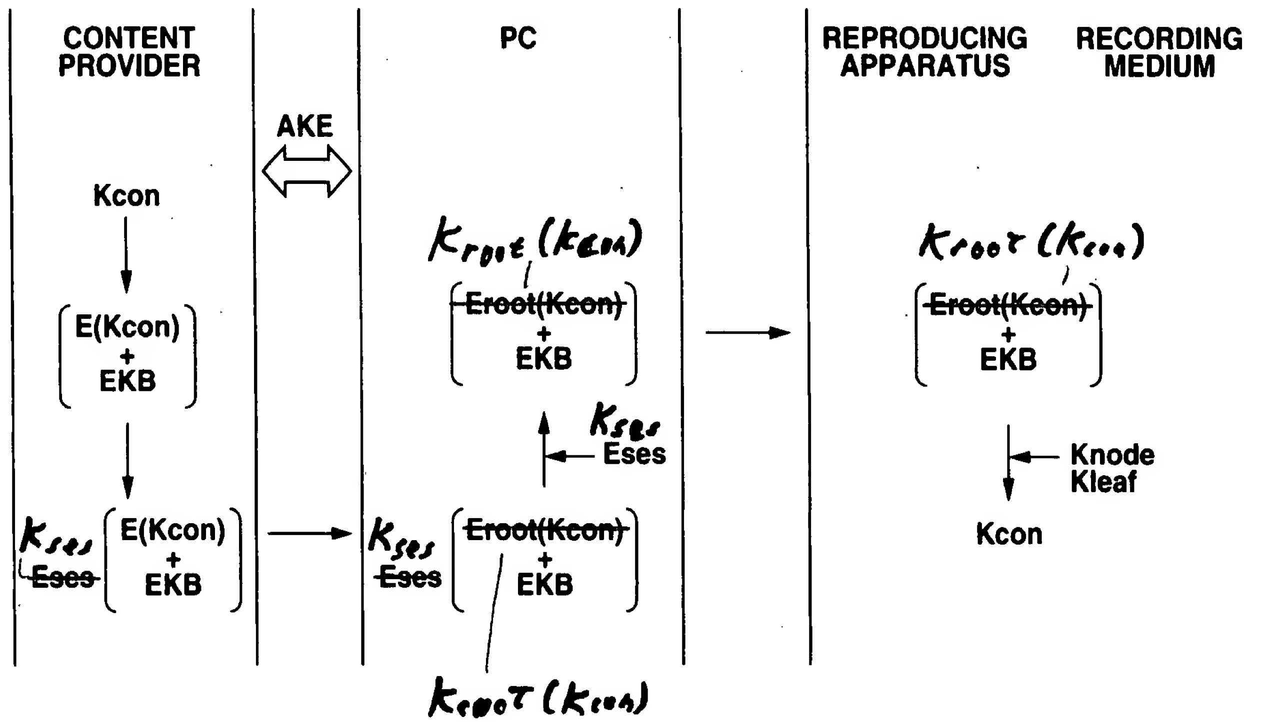
CONTENT PROVIDER
AKE
PC
REPRODUCING APPARATUS
RECORDING MEDIUM
Kcon
E(Kcon) + EKB
Kses
~~Eses~~
E(Kcon) + EKB
Kroot (Kcon)
~~Eroot(Kcon)~~ + EKB
Kses
Eses
Kses
~~Eses~~
~~Eroot(Kcon)~~ + EKB
Kroot (Kcon)
Kroot (Kcon)
~~Eroot(Kcon)~~ + EKB
Knode
Kleaf
Kcon

FIG.16

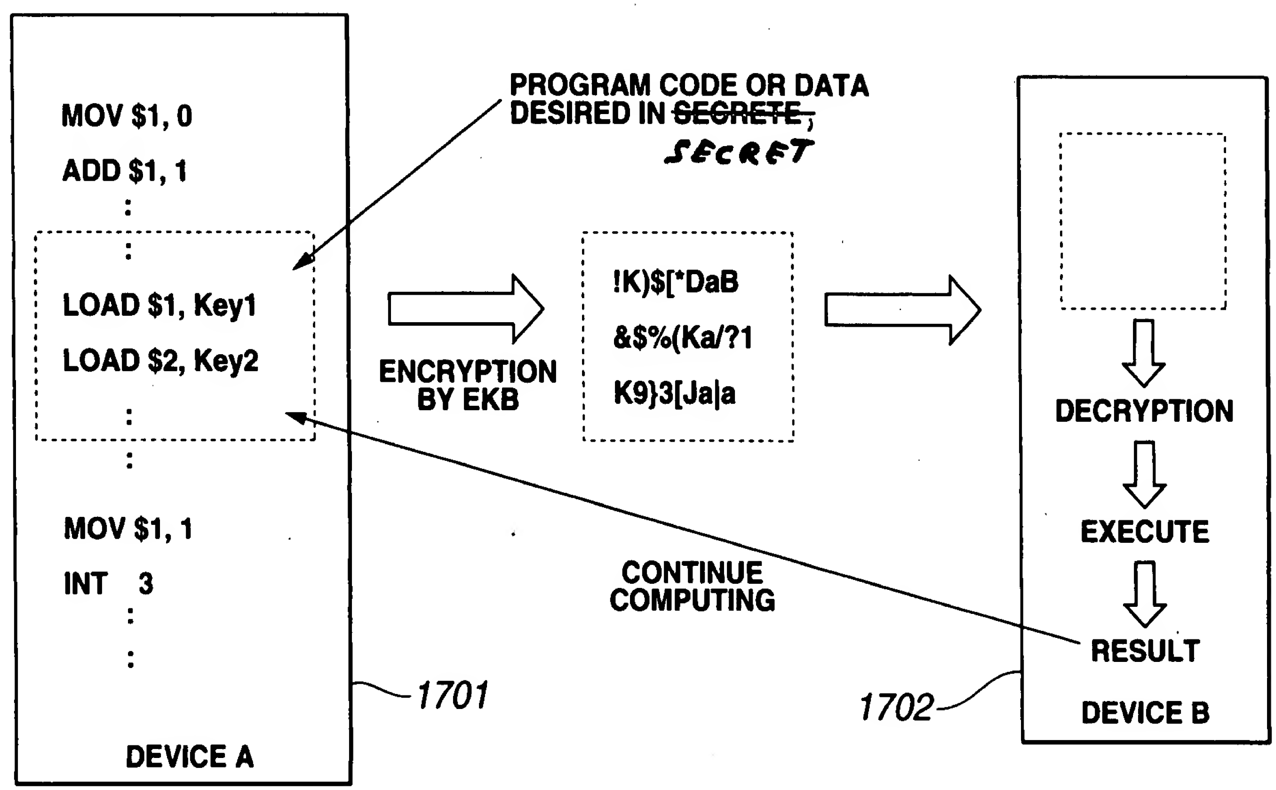
MOV $1, 0
ADD $1, 1
LOAD $1, Key1
LOAD $2, Key2
MOV $1, 1
INT 3
DEVICE A
1701
PROGRAM CODE OR DATA DESIRED IN ~~SECRETE~~, SECRET
ENCRYPTION BY EKB
!K)$[*DaB
&$%(Ka/?1
K9}3[Ja|a
CONTINUE COMPUTING
DECRYPTION
EXECUTE
RESULT
DEVICE B
1702

FIG.17

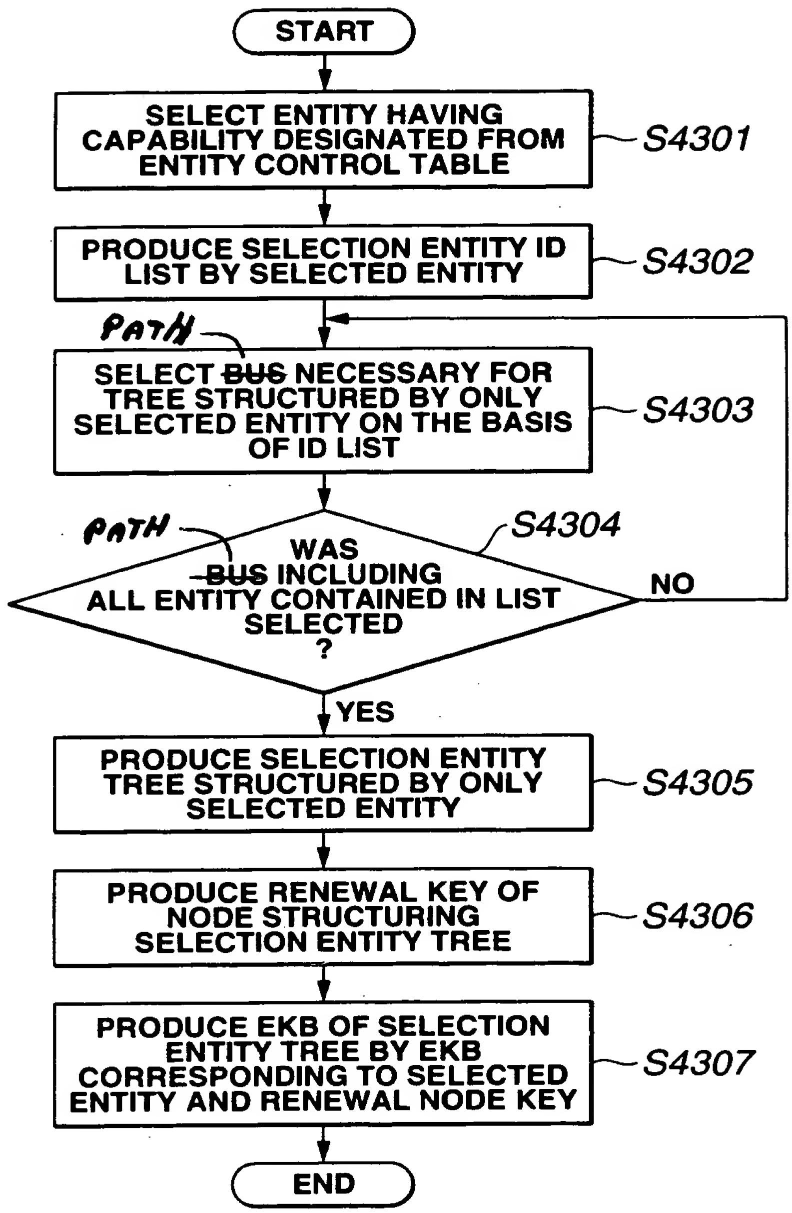
START
SELECT ENTITY HAVING CAPABILITY DESIGNATED FROM ENTITY CONTROL TABLE
S4301
PRODUCE SELECTION ENTITY ID LIST BY SELECTED ENTITY
S4302
PATH
SELECT ~~BUS~~ NECESSARY FOR TREE STRUCTURED BY ONLY SELECTED ENTITY ON THE BASIS OF ID LIST
S4303
S4304
PATH
WAS ~~BUS~~ INCLUDING ALL ENTITY CONTAINED IN LIST SELECTED ?
NO
YES
PRODUCE SELECTION ENTITY TREE STRUCTURED BY ONLY SELECTED ENTITY
S4305
PRODUCE RENEWAL KEY OF NODE STRUCTURING SELECTION ENTITY TREE
S4306
PRODUCE EKB OF SELECTION ENTITY TREE BY EKB CORRESPONDING TO SELECTED ENTITY AND RENEWAL NODE KEY
S4307
END

FIG.44